

Universidad Carlos III de Madrid  
Escuela Politécnica Superior  
Ingeniería Industrial



PROYECTO FIN DE CARRERA

Sistema Web para la predicción  
automática de precios de la  
energía eléctrica

**Autor:** Arturo Soucase Ricau

**Tutores:** Francisco Javier Nogales

Andrés M. Alonso Fernández

**Fecha:** Leganés, noviembre de 2012



# Agradecimientos

A mis padres, a mi hermano, a toda mi familia y en especial a mi pareja.





# Resumen

El consumo eléctrico per cápita es considerado uno de los principales indicadores del nivel de vida de un país o de una región. Tanto la industrialización, un fenómeno que afecta a las regiones menos desarrolladas y que trae consigo niveles de vida más elevados y un crecimiento económico, así como la urbanización, esto es, el aumento de la proporción de población urbana con respecto a la rural, ha provocado un creciente consumo energético a escala mundial.

En España, hasta los años 60, los precios de la energía eléctrica eran estables debido a los bajos costes de capital, una previsible tendencia demográfica y los precios estables de los combustibles. A partir de los años 70, con las crisis del petróleo, la volatilidad en los precios de la energía aumentó por lo que la planificación de la energía eléctrica fue adquiriendo un gran protagonismo. A partir de 1990, se extendió el proceso de desregularización convirtiendo la energía eléctrica en una “commodity” gobernada por la ley de la oferta y la demanda.

El presente documento recoge las distintas tecnologías empleadas, las fases de diseño e implantación, y la documentación del sistema de predicción de precios de la energía eléctrica desarrollado, así como la interfaz web disponible en <http://pricefor.uc3m.es>.





# Índice General

1. Introducción .....	18
1.1 Motivación .....	18
1.2 Objetivo .....	19
1.3 Estructura .....	20
2. Estado del arte.....	22
2.1 Mercado eléctrico español.....	22
2.1.1 Descripción general .....	22
2.1.2 Mercado diario .....	24
2.1.3 Secuencia de operaciones en el mercado diario.....	25
2.2 Información pública .....	25
2.3 Herramienta Matlab de previsión de precios.....	27
3. Tecnologías aplicadas y herramientas de trabajo .....	29
3.1 Sistema operativo.....	29
3.1.1 Microsoft Windows Server 2003.....	30
3.2 Servidores Web .....	31
3.2.1 Protocolo HTTP .....	31
3.2.2 Estructura de métodos y respuestas HTTP/1.0.....	32
3.2.3 TLS / SSL.....	34
3.2.4 Servidores web en el mercado .....	36
3.2.5 PriceForecast Web Server .....	38
3.3 Sistemas de gestión de bases de datos.....	40
3.3.1 Definición .....	40
3.3.2 Modelos .....	40
3.3.3 Sistemas de gestión SQL.....	41
3.3.4 NoSQL.....	42
3.3.5 MongoDB .....	43



3.4	Matlab.....	44
3.4.1	Descripción general .....	44
3.4.2	Matlab Compiler .....	44
3.4.3	Matlab PHP-MySQL Gateway .....	46
3.5	Lenguajes de programación .....	48
3.5.1	Lenguajes compilados e interpretados .....	48
3.5.2	PHP .....	48
3.5.3	JavaScript.....	49
3.5.4	Java .....	49
3.6	XHTML & XML.....	51
3.6.1	XHTML.....	51
3.6.2	XML .....	51
3.7	Librería externas .....	52
3.7.1	Twitter Bootstrap .....	52
3.7.2	jQuery .....	52
3.7.3	pChart.....	52
3.7.4	DropBox PHP Lib .....	52
3.7.5	Google API PHP Client.....	53
3.7.6	PHPExcel.....	53
3.7.7	PHP Mailer .....	53
3.7.8	reCAPTCHA.....	53
3.7.9	Fcgi4j.....	53
3.7.10	NicEdit .....	54
3.8	Web Server Stress Test .....	54
3.8.1	ApacheBench (ab).....	54
3.9	Web-Based Exploitation .....	54
3.9.1	Inyecciones SQL .....	54
3.9.2	Inyecciones NoSQL .....	55
3.9.3	Cross-Site Scripting (XSS).....	56
3.9.4	Nikto .....	56
3.10	Port & Vulnerability Scanning .....	56
3.10.1	Nmap.....	57

3.10.2	Nessus.....	58
3.11	Server Exploitation .....	58
3.11.1	Metasploit Framework.....	59
3.11.2	Fast-Track Autopwn.....	59
3.12	Cortafuegos & antivirus.....	59
4.	Análisis, diseño e implantación del Backend .....	61
4.1	Análisis.....	61
4.1.1	Requisitos funcionales .....	61
4.1.2	Requisitos no funcionales .....	63
4.2	Diseño .....	65
4.2.1	Modelo conceptual .....	65
4.2.2	Modelo de la base de datos.....	68
4.3	Implementación.....	73
4.3.1	Adquisición de datos .....	73
4.3.2	Procesado de datos .....	76
4.3.3	Interpolación de datos.....	77
4.3.4	Generación de predicciones horarias .....	82
4.3.5	Generación de predicciones mensuales y pico.....	83
4.3.6	Gestión de datos de predicciones .....	84
4.3.7	Cálculo de errores acumulados.....	84
4.3.8	Backup .....	84
5.	Análisis, diseño e implementación del Frontend.....	86
5.1	Análisis.....	86
5.1.1	Diagrama de caso de uso.....	86
5.2	Diseño .....	94
5.2.1	Modelo de la base de datos.....	94
5.3	Implementación.....	98
5.3.1	Diseño de la interfaz web .....	98
5.3.2	Funcionalidad general .....	99
5.3.3	Registro de usuarios .....	102
5.3.4	Identificación de usuarios.....	102
5.3.5	Administración de la aplicación web.....	105



6.	API HTTP/RESTful.....	107
6.1	Descripción general .....	107
6.2	Estructura .....	108
6.3	Ejemplos de uso .....	110
6.4	Aplicación “PriceForecasting Client” .....	111
7.	Evaluación.....	116
7.1	MySQL – PHP vs. MySQL ODBC Driver.....	116
7.2	Web Server Stress Tests.....	117
7.2.1	Caso I.....	117
7.2.2	Caso II.....	120
7.2.3	Caso III .....	122
7.3	Web-Based Exploitation .....	124
7.4	Port & Vulnerability Scanning .....	126
7.4.1	Nmap.....	126
7.4.2	Nessus.....	128
7.5	Server Exploitation .....	133
8.	Conclusiones.....	136
9.	Bibliografía.....	138
10.	Planificación.....	140
11.	Presupuesto.....	142
11.1	Coste de personal.....	142
11.2	Hardware.....	143
11.3	Software y Licencias.....	143
11.4	Resumen.....	144

## Índice de Figuras

Figura 1. Secuencia de mercados en el Mercado Ibérico de Electricidad .....	23
Figura 2. Casación de precios horarios .....	24
Figura 3. Secuencia de ejecución y horizontes en el mercado diario .....	25
Figura 4. Resultados del Mercado en OMIE.es .....	26
Figura 5. Sistema de Información ESIOS en REE.es .....	27
Figura 6. Esquema de peticiones en el protocolo HTTP entre cliente y servidor.....	31
Figura 7. Fases en el protocolo TLS .....	35
Figura 8. Evolución de la cuota de mercado de servidores web. ....	36
Figura 9. Esquema de compilación en Matlab.....	46
Figura 10. Diagrama UML de actividades para predicciones horarias.....	66
Figura 11. Diagrama UML de actividades para predicciones mensuales y de pico.....	67
Figura 12. Modelo de base de datos I.....	68
Figura 13. Modelo de base de datos II .....	68
Figura 14. Modelo de base de datos III.....	69
Figura 15. Modelo de la base de datos IV .....	70
Figura 16. Modelo de base de datos V .....	70
Figura 17. Modelo de la base de datos VI.....	71
Figura 18. Modelo de base de datos VII.....	72
Figura 19. Demandas horarias reales 2010-2012 .....	78
Figura 20. Superposición de demandas horarias reales 2010-2012 .....	79
Figura 21. Gráfica de errores MAD. ....	81
Figura 22. Gráfica de errores MSE.....	81
Figura 23. Diagrama de casos de uso.....	87
Figura 24. Página "dashboard" de la interfaz web.....	99
Figura 25. Sección "Gráfico" de la interfaz web.....	100
Figura 26. Sección "Resultados" de la interfaz web .....	100
Figura 27. Sección "Descargas" de la interfaz web.....	101
Figura 28. Sección "Gráfico" en las páginas del grupo "Errores" .....	101
Figura 29. Página "register" de la interfaz web.....	102
Figura 30. Página "login" de la interfaz web.....	103
Figura 31. Página "recover" de la interfaz web .....	103



Figura 32. Menú de usuario .....	104
Figura 33. Página "profile" de la interfaz web .....	104
Figura 34. Administración interfaz web.....	105
Figura 35. Modelo API Cliente-Servidor .....	108
Figura 36. PriceForecast Client.....	111
Figura 37. Interfaz cliente API .....	112
Figura 38. Sección D de "PriceForecast Client" .....	113
Figura 39. Pestaña 'Request' de "PriceForecast Client" .....	114
Figura 40. Pestaña 'Http Headers' de "PriceForecast Client" .....	114
Figura 41. Pestaña 'Raw Data' de "PriceForecast Client" .....	115
Figura 42. Gráfica resultados AB. Apache. Caso I.....	118
Figura 43. Gráfica resultados AB. PriceForecast Web Server. Caso I .....	119
Figura 44. Gráfica resultados AB. Apache. Caso II .....	120
Figura 45. Gráfica resultados AB. Apache. Caso III.....	122
Figura 46. Nessus. Análisis básico .....	128
Figura 47. Nessus. Análisis avanzado I.....	129
Figura 48. Nessus. Análisis avanzado I (continuación).....	130
Figura 49. Nessus. Análisis avanzado II.....	131
Figura 50. Nessus. Análisis avanzado II (continuación) .....	132





## Índice de Tablas

Tabla 1. Cuota de mercado de servidores web. Septiembre 2012.....	36
Tabla 2. RF-01 .....	61
Tabla 3. RF-02 .....	61
Tabla 4. RF-03 .....	62
Tabla 5. RF-04 .....	62
Tabla 6. RF-05 .....	62
Tabla 7. RF-06 .....	62
Tabla 8. RF-07 .....	62
Tabla 9. RF-08 .....	63
Tabla 10. RF-09 .....	63
Tabla 11. RF-10 .....	63
Tabla 12. RFN-01.....	63
Tabla 13. RFN-02.....	64
Tabla 14. RFN-03.....	64
Tabla 15. RFN-04.....	64
Tabla 16. RFN-05.....	64
Tabla 17. RFN-06.....	64
Tabla 18. RFN-07.....	65
Tabla 19. RFN-08.....	65
Tabla 20. RFN-09.....	65
Tabla 21. Errores de predicción. ....	80
Tabla 22. CU-01 .....	88
Tabla 23. CU-02 .....	88
Tabla 24. CU-03 .....	89
Tabla 25. CU-04 .....	89
Tabla 26. CU-05 .....	90
Tabla 27. CU-06 .....	90
Tabla 28. CU-07 .....	91
Tabla 29. CU-08 .....	91
Tabla 30. CU-09 .....	92
Tabla 31. CU-10 .....	92
Tabla 32. CU-11 .....	93



Tabla 33. CU-12 .....	93
Tabla 34. CU-12 .....	94
Tabla 35. Resultados tests PHP-MySQL.....	116
Tabla 36. Resultados tests MySQL ODBC Driver .....	117
Tabla 37. Tabla de resultados AB. Apache. Caso I.....	118
Tabla 38. Tabla de resultados AB. PriceForecast Web Server. Caso I.....	119
Tabla 39. Tabla de resultados AB. Apache. Caso II .....	121
Tabla 40. Tabla de resultados AB. Apache. Caso III .....	122
Tabla 41. Coste de personal .....	142
Tabla 42. Detalle de hardware .....	143
Tabla 43. Coste software y licencias .....	143
Tabla 44. Resumen costes .....	144

## Glosario de Términos

- **AB:** Apache Benchmark.
- **AES:** Advanced Encryption Standard.
- **API:** Application Programming Interface.
- **BD:** Base de datos.
- **BOE:** Boletín Oficial del Estado.
- **BSON:** Binary JSON.
- **CGI:** Common Gateway Interface.
- **CPU:** Central Processor Unit.
- **CSS:** Cascade Style Sheet.
- **DOM:** Document Object Model.
- **E/S:** Entrada y salida.
- **GB:** GigaByte.
- **GNU:** GNU is Not Unix.
- **GPL:** GNU General Public License.
- **GUI:** Graphical User Interface.
- **HTTP:** Hypertext Transfer Protocol.
- **IDE:** Integrated Development Environment.
- **IIS:** Internet Information Services.
- **IMAP:** Internet Message Access Protocol.
- **JSON:** JavaScript Object Notation.
- **MB:** MegaByte.
- **MCR:** Matlab Compiler Runtime.
- **MIBEL:** Mercado Ibérico de Electricidad.
- **OMIE - OMEL:** Operador del Mercado Ibérico de Energía – Polo Español, S.A.
- **OSI:** Open Systems Interconnection.
- **OTC:** Over the Counter
- **RAM:** Random-Access Memory.
- **RDBMS:** Relational Database Management System



- **REE:** Red Eléctrica Española, S.A.
- **REST:** Representational State Transfer.
- **RFC:** Request for Comments.
- **SGML:** Standard Generalized Markup Language.
- **SMTP:** Simple Mail Transfer Protocol.
- **SO:** Sistema operativo.
- **SOAP:** Simple Object Access Protocol.
- **SQL:** Structured Query Language.
- **SSL:** Secure Sockets Layer.
- **TLS:** Transport Layer Security.
- **TUR:** Tarifa de Último Recurso.
- **URI:** Uniform Resource Identifier.
- **URL:** Universal Resource Locator.
- **VM:** Virtual Machine.
- **WWW:** World Wide Web.
- **XHTML:** Extensible Hypertext Markup Language.
- **XSS:** Cross-Site Scripting.



# Capítulo 1

## Introducción

### 1.1 Motivación

El suministro de energía eléctrica es esencial en el funcionamiento de las sociedades modernas, ya que se considera un bien de primera necesidad. La industria eléctrica, cuyo origen se remonta a 1873, se caracteriza por suministrar un servicio básico a los ciudadanos y las empresas. Por lo tanto, el objetivo fundamental de cualquier política energética de un país es el asegurar un suministro constante y completo de energía eléctrica.

El sector eléctrico en España ha experimentado importantes cambios en las últimas décadas. Se ha transformado un sistema completamente regulado e intervenido por los poderes públicos, a un sistema liberalizado con la entrada en operación de los mercados eléctricos.

El proceso de liberalización comenzó en España en 1997 con la aprobación de la Ley 54/1997, conocida como la Ley del Sector Eléctrico. En 1998 entró en funcionamiento el mercado liberalizado de la energía, aunque solo un reducido número de usuarios cualificados pudieron beneficiarse de él. A partir del 1 de enero de 2003, la legislación española reconoce al consumidor de energía el derecho a elegir la forma en la que contrata su suministro eléctrico. Finalmente, el 1 de julio de 2009 el mercado eléctrico español quedó completamente liberalizado con la entrada en vigor de la Tarifa de Último Recurso (TUR).

El objetivo de este marco legislativo es aumentar la competitividad y reducir los precios. La TUR, a su vez, ha permitido que el suministro de energía eléctrica siga siendo un servicio básico y asequible.

La entrada en operación de los mercados competitivos hace necesaria la predicción de los precios de la energía eléctrica. Esta predicción es imprescindible para que los productores y comercializadores puedan elaborar sus ofertas diarias.

La liberalización del sector también ha permitido que la energía eléctrica pase a ser considerada una “commodity”, por lo que han ido formándose nuevos mercados “Over the Counter” (OTC) de energía eléctrica en España. En ellas, algunas empresas compran y venden energía y derivados financieros al por mayor, y otras intermedian las compraventas de energía y derivados financieros sobre estos mercados. Se estima que el mercado de derivados a nivel global pueda alcanzar entre 5.000 y 10.000 TWh al año. Estas empresas requieren de predicciones de precios de la electricidad para realizar el “trading” y gestionar riesgos.

## 1.2 Objetivo

El objetivo de este proyecto ha sido llevar a cabo el análisis, diseño e implantación de un sistema completo que realice las siguientes funciones:

- Conexiones diarias con los servidores web de Operador del Mercado Ibérico de Energía – Polo Español, S.A. (OMIE) y Red Eléctrica Española, S.A. (REE) para la descarga de precios y demandas horarias programadas, así como las predicciones de demandas mensuales con un horizonte anual.
- Enlace con las herramientas estadísticas para generar predicciones de precios de energía eléctrica introduciendo como inputs las variables antes mencionadas y la planificación de días festivos.
- Ejecución en un servidor de una herramienta con una metodología definida para predecir precios a distintos horizontes prefijados.
- Evaluación sistemática del desempeño de los modelos desarrollados. Dado un modelo, generar automáticamente errores pasados con distintos horizontes de predicción.
- Generar de forma automática diversos outputs en un servidor web tales como predicciones, trayectorias, bandas de confianza y medidas de error con distintos horizontes.

Las predicciones de precios de energía eléctrica que se obtendrán son horarias, mensuales y pico. Los precios de horas pico son aquellos comprendidos entre las 8:00 horas de la mañana y las 20:00 horas de la tarde.

La implantación del sistema requirió la compra de un servidor con las siguientes especificaciones:

- **Procesador:** Intel Core 2 Quad Q9400 2.66 GHz.
- **Memoria RAM:** 4 GB.
- **Disco Duro:** 500 GB.
- **Sistema Operativo:** Windows Server 2003.

Se determinó que para llevar a cabo de forma exitosa el proyecto, éste debía cumplir una serie de principios básicos:

- **Portabilidad:** El sistema debe ser completamente independiente del entorno en el que se ejecuta. Para ello se utilizan tecnologías “open source” y multiplataforma.
- **Estabilidad:** El sistema debe ser robusto ante un elevado número de peticiones.
- **Seguridad:** El servidor en el que se ejecuta el proyecto debe estar protegido frente a ataques que permitan un acceso remoto al mismo y a los datos y herramientas desarrolladas.
- **Mínima administración y mantenimiento:** El sistema debe ser completamente autónomo en condiciones normales.

### 1.3 Estructura

Este documento está estructurado en ocho capítulos:

- En el primer capítulo, “Introducción”, se presenta una descripción general del proyecto y los objetivos del mismo.
- En el segundo capítulo, “Estado del arte”, se describe el funcionamiento actual del mercado eléctrico en España, haciendo especial hincapié en el mercado diario y su secuencia de operaciones.
- En el tercer capítulo, “Tecnología aplicadas y herramientas de trabajo”, se detallan todo el conjunto de tecnologías utilizadas, como lenguajes de programación y librerías, y herramientas de trabajo, como Matlab y Apache HTTP Server.
- En el cuarto capítulo, “Análisis, diseño e implantación del Backend”, se describen con detalle los requisitos funcionales de los que debe disponer el proyecto, el modelo de base de datos empleado y la implementación final del ‘backend’.
- En el quinto capítulo, “Análisis, diseño e implantación del Frontend”, se describe, mediante diagramas de caso de uso, la funcionalidad y comportamiento de la interfaz web, y la implementación final del ‘frontend’.



- En el sexto capítulo, “API HTTP/RESTful”, se describe el funcionamiento de la API de la que dispone el proyecto y una aplicación informática de ejemplo desarrollado para la misma.
- En el séptimo capítulo, “Evaluación”, se detallan las distintas pruebas que se han ejecutado en el sistema para comprobar su estabilidad y seguridad y los resultados obtenidos. Además, se indican las medidas correctivas aplicadas en base a los resultados anteriores.
- Por último, en el octavo capítulo, “Conclusiones”, se comentan los resultados de la evaluación y, como mediante estos, podemos concluir que se han cumplido los objetivos propuestos en el primer capítulo.

## Capítulo 2

### Estado del arte

#### 2.1 Mercado eléctrico español

##### *2.1.1 Descripción general*

Tras el inicio de la liberalización del sector eléctrico en España, la Ley del Sector Eléctrico definió una serie de agentes participantes que debían desarrollar actividades específicas en el nuevo mercado. Entre ellas destacan:

- **Generadores:** Son aquellas personas o entidades físicas o jurídicas cuya función es la de generar energía eléctrica, así como construir, operar y mantener las centrales de generación.
- **Red de transporte:** Son sociedades mercantiles que transportan la electricidad desde los centros de producción hasta las redes de distribución, y además, construyen, mantienen y operan las instalaciones de la red de transporte.
- **Distribuidores:** Son sociedades mercantiles cuya función es situar la energía en los puntos de consumo, así como desarrollar, mantener y operar las instalaciones necesarias para tal fin.
- **Comercializadores:** Empresas con acceso a las redes de transporte y distribución que venden energía eléctrica a los consumidores o a otros comercializadores.
- **Regulador:** Ente público que vela por el cumplimiento de la regulación básica y determina los requisitos mínimos de calidad y seguridad en el suministro eléctrico.
- **Operador del sistema:** Es el responsable de la gestión técnica del sistema. Su objetivo es garantizar la continuidad y seguridad del suministro.
- **Operador del mercado:** Es el responsable de la gestión económica del sistema. Gestiona el sistema de ofertas de compra y venta de energía que los diferentes agentes efectúan en el mercado eléctrico.

Existen dos estructuras de mercado para facilitar el intercambio de energía entre productores y consumidores: la bolsa de energía o “pool” y los contratos bilaterales.



Figura 1. Secuencia de mercados en el Mercado Ibérico de Electricidad

Fuente: <http://www.energiaysociedad.es>

En el “pool”, los productores presentan al operador del mercado las ofertas de producción. Éstas están compuestas de un conjunto de bloques de energía eléctrica para cada una de las veinticuatro horas del día siguiente y el precio mínimo de venta de cada bloque. Análogamente, los comercializadores y grandes consumidores presentan al operador del mercado sus ofertas de compra de energía eléctrica, compuestas de bloques y el precio máximo de compra de cada bloque, para el mismo horizonte temporal horario.

Un mecanismo de casación establece el precio horario de la energía así como la energía que producirá cada productor y comprará cada comercializador o gran consumidor. Aquellas ofertas de venta de energía eléctrica con un precio inferior al precio de cierre del mercado son aceptadas así como las ofertas de compra con un precio superior. Las ofertas con un precio igual al precio de cierre del mercado son procesadas proporcionalmente en función de la energía ofertada. Las ofertas restantes son rechazadas.

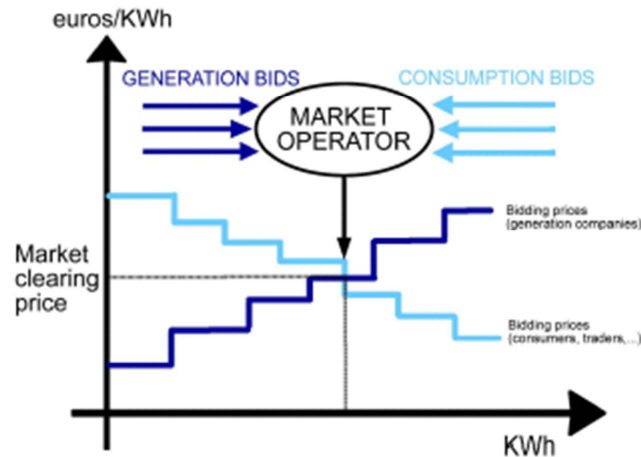


Figura 2. Casación de precios horarios

Por otro lado, los contratos bilaterales se realizan entre consumidores, comercializadores y productores de energía. Estos contratos se comunican al operador del mercado para ser tenidos en cuenta en la realización del algoritmo de cierre del mercado.

Finalmente el operador del mercado informa al operador del sistema acerca del programa diario de producción y este último debe asegurar la seguridad del sistema. El operador del sistema incluso puede introducir nuevos cambios si lo considera necesario.

### 2.1.2 Mercado diario

El mercado diario es parte integrante del mercado de energía eléctrica. Su objetivo es llevar a cabo las transacciones de energía eléctrica para el siguiente día presentadas por los agentes del mercado.

El precio que tiene en el mercado diario está basado en la intersección de la curva de oferta con la curva de demanda, formadas a partir de las ofertas de venta y compra hasta alcanzar el punto de equilibrio del mercado.

### 2.1.3 Secuencia de operaciones en el mercado diario

La casación de ofertas y demandas para cada día comienza el día anterior al mismo.

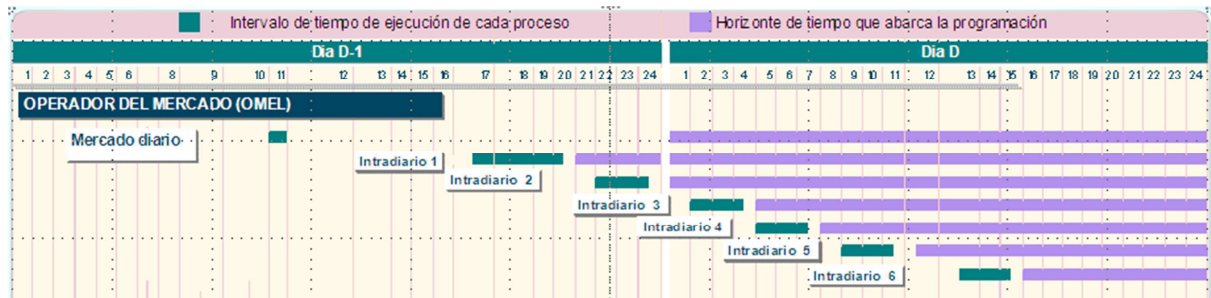


Figura 3. Secuencia de ejecución y horizontes en el mercado diario

Fuente: <http://www.energiaysociedad.es>

El operador del mercado admite las ofertas que recibe hasta la hora de cierre, las 10:00h. Entorno a las 10:30h se publica el resultado de la casación, es decir, el programa base de funcionamiento diario, que está constituido por las operaciones en el mercado diario y los contratos bilaterales.

Publicado el programa base, se analizan y se incorporan las modificaciones necesarias para resolver las restricciones técnicas identificadas por criterios de seguridad y a las 14:00h se publica el programa diario viable provisional. Se procede a continuación a la subasta de la banda de regulación secundaria, haciéndose público el programa diario viable a las 16:00h.

Tras la publicación del programa diario viable, se abre el mercado intradiario que genera finalmente el programa horario final.

## 2.2 Información pública

En cumplimiento de las disposiciones del Real Decreto-Ley 6/2000 y de las resoluciones del Secretario de Estado de Economía, de la Energía y de la Pequeña y Mediana Empresa de cinco de abril de 2001 y de 10 de mayo de 2001, el operador del mercado publica los resultados de los mercados y las liquidaciones del mercado de producción en su página web, <http://www.omie.es>.

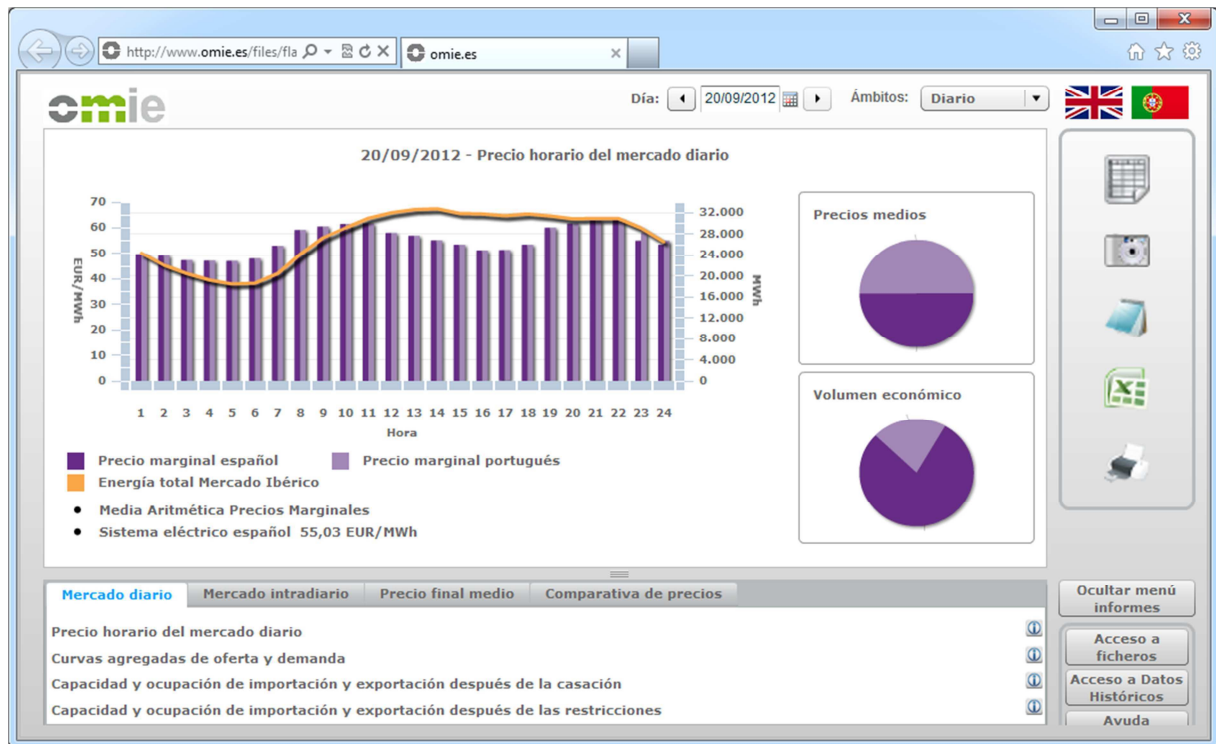


Figura 4. Resultados del Mercado en OMIE.es

Fuente: <http://www.omie.es>

Los datos diarios publicados derivados del proceso de casación del mercado diario para MIBEL, desglosados el sistema eléctrico español del portugués son:

- **Precios horarios:** fichero 'marginal\_pdbc'.
- **Energía total negociada horaria del mercado diario:** fichero 'pdbc\_tot'.
- **Curvas agregadas de venta y adquisición, y curvas de programa resultante de casación de venta y adquisición:** fichero 'curva\_pbc'.
- **Ocupación de cada una de las interconexiones internacionales por hora:** fichero 'capacidad\_inter\_pbc'.

Con respecto al operador del sistema, en cumplimiento de las resoluciones del Secretario de Estado de Economía de 27 de octubre de 2010, tiene la obligación de hacer públicos los resultados de los mercados o procesos de operación del sistema.

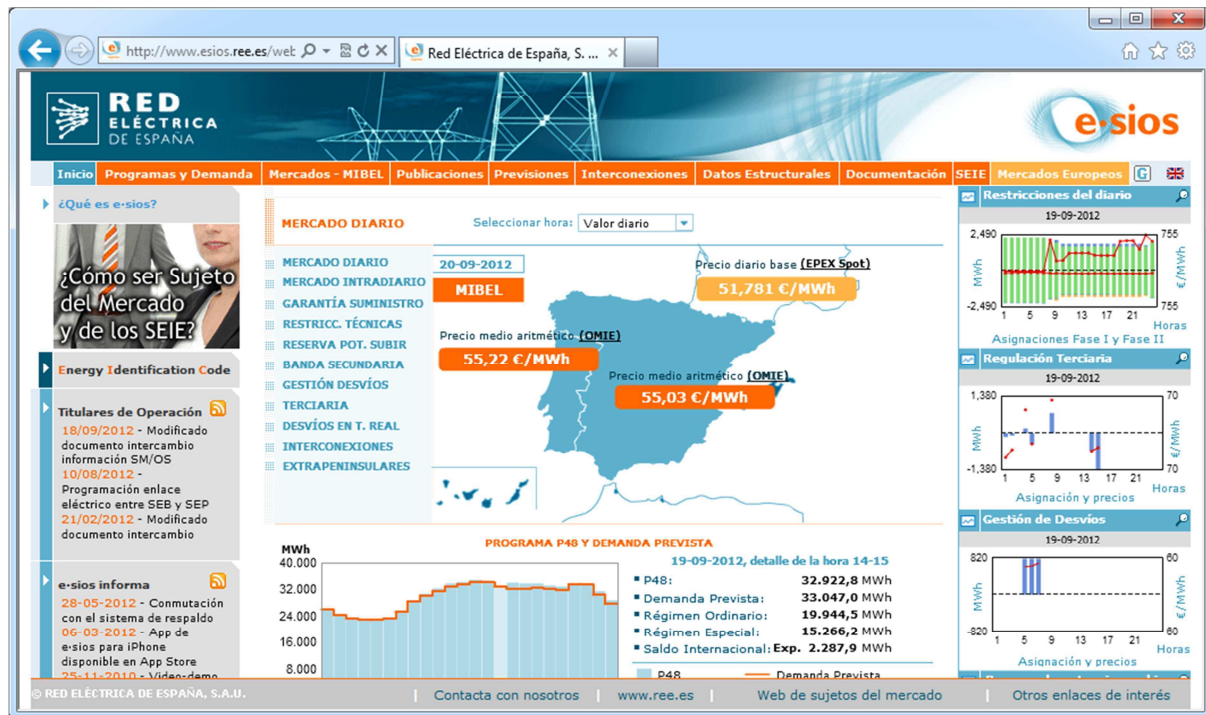


Figura 5. Sistema de Información ESIOs en REE.es

Fuente: <http://www.esios.ree.es>

Para ello REE ha desarrollado un sistema de información denominado ‘Sistema de información del operador del sistema E·SIOs’. En él se desarrollan las comunicaciones entre OMIE, los sujetos del mercado y los propios técnicos del operador, y además publica los resultados de los distintos mercados y programaciones.

## 2.3 Herramienta Matlab de previsión de precios

Este proyecto realiza las predicciones de precios en base a una herramienta informática desarrollada por los tutores del mismo.

Esta herramienta denominada PriceForecastUC3M ha sido desarrollada en Matlab y distribuida al autor del proyecto en formato P-CODE. Este formato propietario es exclusivo de Matlab. Permite crear una función protegida con el algoritmo de cifrado AES con una longitud de clave no documentada. Por lo tanto, el cometido principal de este formato es la protección de la propiedad intelectual mediante la ofuscación del código fuente.



El uso de una rutina cifrada en formato P-CODE frente a la misma no cifrada, no genera una disminución en la velocidad de ejecución significativa, por lo que su uso a nivel de producción es válido.

Incluye así mismo la posibilidad de incorporar su código como librería a cualquier aplicación “standalone” desarrollada en Matlab de forma transparente.



## Capítulo 3

# Tecnologías aplicadas y herramientas de trabajo

### 3.1 Sistema operativo

Un sistema operativo (SO) se define como un programa que facilita el uso del soporte físico, por lo general escrito en lenguaje de alto nivel, diseñado de forma que sea fácil de comprender y sencillo de utilizar. Su principal cometido es la de actuar como intermediario entre los programas de aplicación y el soporte físico. El SO se interpone entre el usuario y el soporte físico de manera que el usuario no precisa conocer los detalles de funcionamiento del soporte en el que se ejecuta, de los dispositivos de almacenamiento y de los dispositivos de entrada y salida (E/S).

Los principales componentes de un SO son:

- **Gestión de procesos:** Los procesos son programas en ejecución que requieren recursos tales como CPU, memoria y acceso a ficheros.
- **Gestión de memoria:** El gestor de memoria asigna una cantidad de memoria específica a cada proceso que lo solicita y lo retira cuando el proceso ha terminado o está temporalmente suspendido de ejecución.
- **Gestión de ficheros:** El SO se hace cargo de crear, gestionar y manipular ficheros y directorios de forma transparente.
- **Gestión de recursos:** Otros recursos de los que se hace cargo el SO son la gestión de E/S y los sistemas de comunicaciones y redes.

Cabe destacar que un SO puede estar destinado para ordenadores personales o para servidores. Un ordenador personal es aquel diseñado para realizar trabajos sencillos de oficina y de hogar. Un servidor, en cambio, es una máquina informática que forma parte de una red de ordenadores y proporciona servicios a otros ordenadores denominados clientes.

Debido a que el uso de recursos y hardware es mayor en servidores que en ordenadores personales, existen en el mercado SO diseñados específicamente para cada uno de ellos.

### ***3.1.1 Microsoft Windows Server 2003***

Microsoft Windows Server 2003 es un sistema operativo destinado a servidores basado en el núcleo de Windows NT. Es un SO de código propietario desarrollado por Microsoft.

Windows Server 2003 fue lanzado en el año 2003 y se considera una versión de Windows XP modificado, al que se le han añadido servicios propios de un servidor, y eliminado características para aumentar su rendimiento.

Este sistema operativo destinado a servidores contiene características tales como:

- **Automated System Recovery:** Permite recuperar el sistema operativo a un estado anterior.
- **Volume Shadow Copy:** Ayuda a recuperar ficheros eliminados por error ya que guarda versiones anteriores del mismo para su posterior recuperación.
- **Reboot Reason Collector:** Herramienta para recopilar las causas que han forzado a que el servidor se reinicie o se apague.

Windows Server 2003 se considera un SO fácil de usar ya que provee de un entorno gráfico y es aconsejable para entornos empresariales que no dispongan de administradores para realizar el mantenimiento del servidor.

En comparación con otros SO para servidores como UNIX, desarrollado en los laboratorios de Bell de AT&T en 1969 y que ha evolucionado de forma independiente hasta la actualidad; o Linux, cuyo núcleo de tipo UNIX es de código libre desarrollado por Linus Torvalds en 1991; Windows Server 2003 presenta claras desventajas ya que no tiene a disposición su código fuente, su adaptación y parametrización a proyectos es limitada y poco flexible, su rendimiento en servidores es menor debido a su interfaz gráfica y el coste de las licencias es muy elevado en comparación con UNIX o Linux, que al ser software de código libre, es gratuito.

Cabe destacar en cuanto a seguridad, que el último “Service Pack” para Windows Server 2003 fue lanzado en marzo de 2007 y Microsoft proveerá de actualizaciones de seguridad hasta el 14 de julio de 2015, fecha en la cual finalizará su soporte. Cualquier vulnerabilidad detectada pasada esta fecha no será resuelta y el servidor se considerará inseguro.

## 3.2 Servidores Web

Un servidor web es un programa informático cuya principal función es la de transmitir datos a los clientes solicitantes mediante el protocolo HTTP, que pertenece a la capa de aplicación del modelo OSI.

El programa es una aplicación ejecutada en el servidor que realiza conexiones tanto bidireccionales como unidireccionales de forma síncrona o asíncrona con el cliente que recibe la respuesta a la petición que ha realizado.

### 3.2.1 Protocolo HTTP

El protocolo HTTP, usado en las transacciones de la WWW, fue desarrollado por el World Wide Web Consortium y el Internet Engineering Task Force y publicado la versión HTTP 1.0 en la 'Request for Comments' (RFC) 1945 y la versión última 1.1 en la RFC 2616. Se le asigna habitualmente el puerto TCP 80.

HTTP define la sintaxis y semántica en las comunicaciones entre el software que integra la arquitectura web. Es un protocolo orientado a las transacciones y sigue el esquema de petición-respuesta o más concretamente, cliente-servidor.

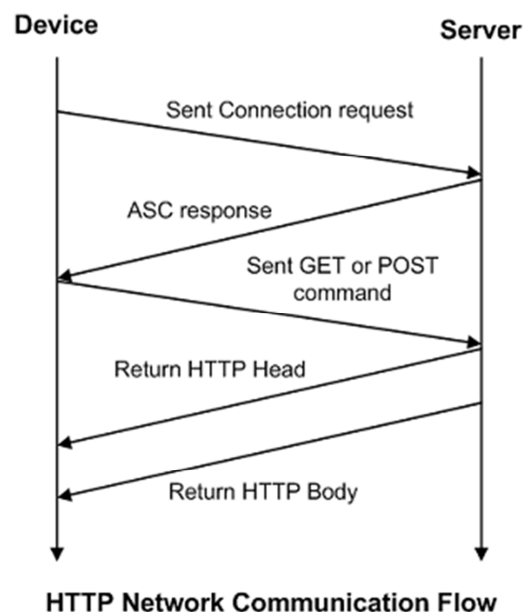


Figura 6. Esquema de peticiones en el protocolo HTTP entre cliente y servidor

Fuente: <http://developer.mediatek.com>

El modelo cliente-servidor es uno de los más empleados entre una aplicación cliente que solicita acceder a los recursos situados en el servidor u ofrecido por una aplicación en el mismo.

HTTP es un protocolo sin estado, es decir, no guarda información alguna sobre las conexiones anteriores que se han realizado. Debido a que las aplicaciones web necesitan frecuentemente mantener el estado, se han desarrollado las denominadas “cookies”, ficheros que se almacenan en los navegadores web, utilizadas para almacenar información de sesión e identificar usuarios.

El protocolo HTTP define una serie de métodos que indican la acción y recurso que solicita el cliente al servidor.

En HTTP/1.0 se definen los siguientes métodos:

- **GET:** Solicita un recurso mediante un Uniform Resource Identifier (URI). El URI es la forma normalizada de referenciar un recurso. A diferencia de un Universal Resource Locator (URL) un URI también contiene el tipo de servicio al que se quiere acceder, la dirección del nodo servidor y el puerto en el que el servicio espera las solicitudes.
- **POST:** Solicita un recurso e incluye información adicional en el cuerpo de la solicitud.
- **HEAD:** Solicita la cabecera del recurso referenciado.

En HTTP/1.1 se añaden cinco métodos adicionales:

- **OPTIONS:** Devuelve los métodos del protocolo que soporta el servidor web dada una URL.
- **PUT:** Carga de un recurso especificado.
- **DELETE:** Borrado del recurso especificado.
- **TRACE:** Solicita un mensaje de respuesta a partir del mensaje de solicitud.
- **CONNECT:** Permite conocer si el host está accesible.

### ***3.2.2 Estructura de métodos y respuestas HTTP/1.0***

A continuación se detalla la estructura típica de los métodos especificados en el protocolo HTTP/1.0 mediante ejemplos y sus respuestas: 200 (petición procesada correctamente), 404 (recurso no disponible), 500 (error genérico) y 501 (método no implementado).

### 3.2.2.1 *Petición GET*

```
GET / HTTP/1.1
Host: pricefor.uc3m.es
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0)
Gecko/20100101 Firefox/14.0.1
Accept: text/html, application/xhtml+xml, application/xml; q=0.9,
*/*; q=0.8
Accept-Encoding: gzip, deflate
Connections: keep-alive
<línea en blanco>
```

### 3.2.2.2 *Petición POST*

```
POST / HTTP/1.1
Host: pricefor.uc3m.es
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0)
Gecko/20100101 Firefox/14.0.1
Accept: text/html, application/xhtml+xml, application/xml; q=0.9,
*/*; q=0.8
Accept-Encoding: gzip, deflate
Connections: keep-alive
<línea en blanco>
Datos
```

### 3.2.2.3 *Petición HEAD*

```
HEAD / HTTP/1.1
Host: pricefor.uc3m.es
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:14.0)
Gecko/20100101 Firefox/14.0.1
Accept: text/html, application/xhtml+xml, application/xml; q=0.9,
*/*; q=0.8
Accept-Encoding: gzip, deflate
Connections: keep-alive
<línea en blanco>
```

### 3.2.2.4 *Respuesta 200*

```
HTTP/1.0 200 OK
Date: Fri Aug 31 17:25:02 CEST 2012
Server: PriceForecastWebServer 1.0
Connection: close
Set-Cookie: PHPSESSID=00dfdhae5786488; path=/
Content-Length: 55844
X-Powered-By: PHP/5.4.4
<línea en blanco>
Datos
```

### 3.2.2.5 *Respuesta 404*

```
HTTP/1.0 404 Not Found
Date: Fri Aug 31 17:25:02 CEST 2012
Server: PriceForecastWebServer 1.0
Connection: close
<línea en blanco>
Datos (Opcional)
```

### 3.2.2.6 *Respuesta 500*

```
HTTP/1.0 500 Internal Server Error
Date: Fri Aug 31 17:25:02 CEST 2012
Server: PriceForecastWebServer 1.0
Connection: close
<línea en blanco>
```

### 3.2.2.7 *Respuesta 501*

```
HTTP/1.0 501 Not Implemented
Date: Fri Aug 31 17:25:02 CEST 2012
Server: PriceForecastWebServer 1.0
Connection: close
<línea en blanco>
```

## 3.2.3 *TLS / SSL*

El Transport Layer Security (TLS), sucesor de Secure Sockets Layer (SSL), es un protocolo de cifrado que proporciona una canal de comunicación segura entre cliente y servidor. Se le asigna habitualmente el puerto TCP 443.

HTTP se considera inseguro ya que está sujeto a ataques “man-in-the-middle”, en el que el atacante tiene capacidad de leer y modificar los datos que se intercambian sin que ni el emisor ni el receptor se percaten de ello. El uso del HTTPS (HTTP cifrado con TLS) permite resistir estos ataques y aumenta la seguridad de las comunicaciones.

El protocolo TLS fue desarrollado por Netscape Communications Corporation y en 1996 se publicó la versión 3.0 utilizado como base para definir el protocolo TLS en la RFC 2246.

Las fases básicas definidas en el protocolo son:

- Negociación del algoritmo usado para la comunicación.
- Intercambio de claves públicas y autenticación basado en certificados digitales.
- Cifrado del tráfico.

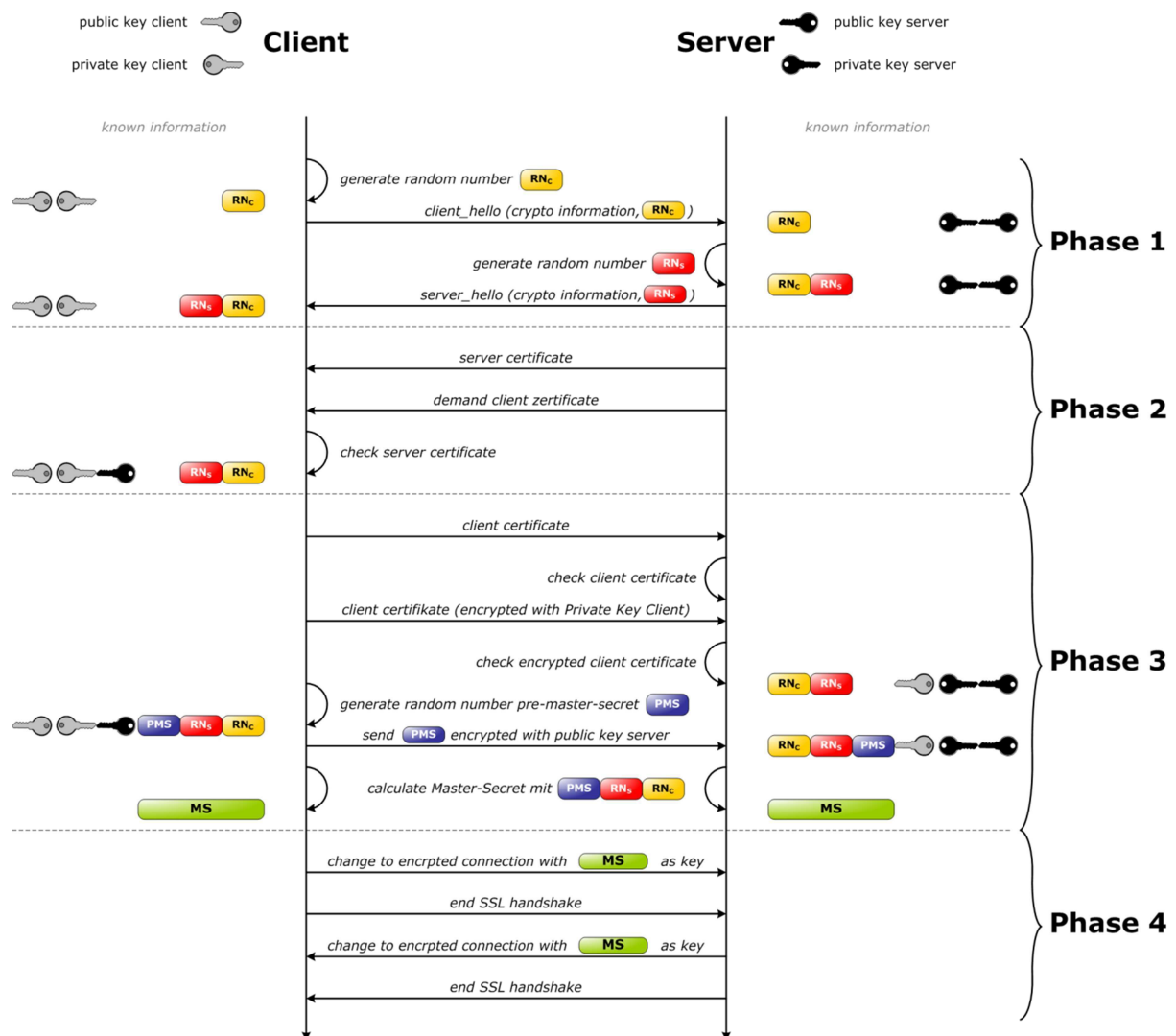


Figura 7. Fases en el protocolo TLS

Fuente: <http://www.hadispatcher.com>

TLS se ejecuta encima de cualquiera de los protocolos en la capa de transporte del modelo OSI, en la capa de sesión. En principio proporciona seguridad en las capas del OSI superiores a la capa de sesión, pero se emplea habitualmente con el protocolo HTTP para formar HTTPS.

El uso de HTTPS en un servidor requiere un certificado de clave pública. Este certificado debe estar firmado por una autoridad de certificación que acredite su autenticidad. La mayoría de navegadores web contienen un listado de certificados raíz firmados por la mayoría de autoridades de certificación.

### 3.2.4 Servidores web en el mercado

Existen una amplia variedad de servidores web disponibles en el mercado. A continuación se muestra una tabla con los servidores web más usados en la actualidad.

Servidor	Desarrollador	Cuota de mercado
Apache	Apache	58,49 %
IIS	Microsoft	15,70 %
Nginx	Nginx	11,93 %
GWS	Google	3,48 %

Tabla 1. Cuota de mercado de servidores web. Septiembre 2012.

Fuente: <http://www.netcraft.com>

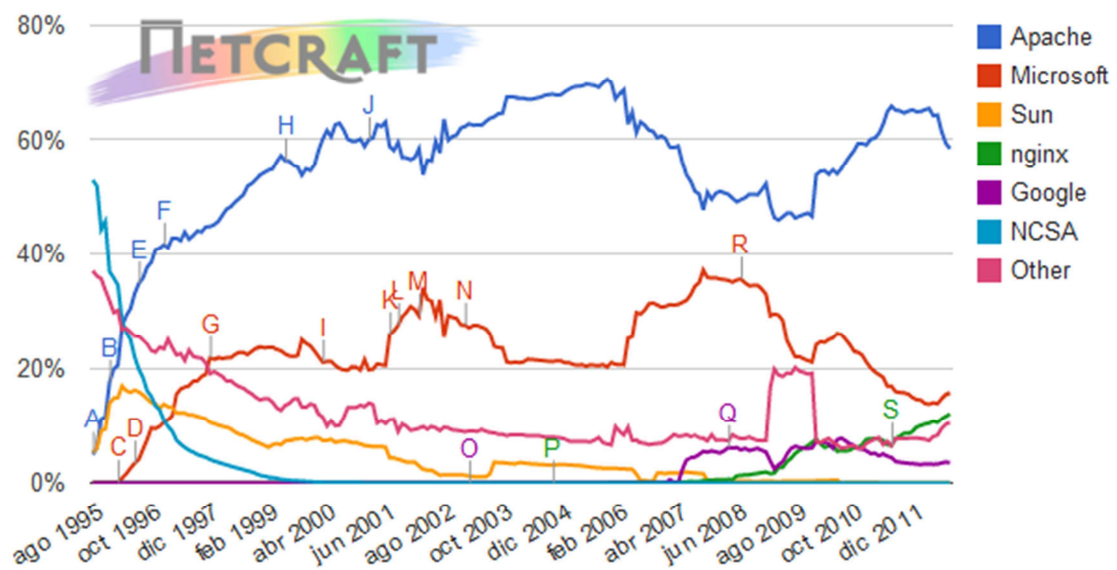


Figura 8. Evolución de la cuota de mercado de servidores web.

Fuente: <http://www.netcraft.com>

Las soluciones disponibles en el mercado para Windows son: IIS, Apache, Nginx y Lighttpd.



#### **3.2.4.1 IIS**

Internet Information Services (IIS) es un servidor web desarrollado por Microsoft para ordenadores que ejecutan Windows. Inicialmente formaba parte como “Option Pack” para Windows NT y posteriormente pasó a formar parte del sistema operativo a partir de Windows 2000.

La versión de IIS incluida en Windows Server 2003 es la 6.0. Los servicios que ofrece esta versión son: FTP, SMTP, NNTP y HTTP/HTTPS. En la versión 6.0 se incluyó soporte para IPv6 y un nuevo administrador de procesos para un mayor rendimiento y seguridad.

La versión posterior, la 7.0 solo está disponible para versiones de Windows Server posteriores a la versión 2003. Como el periodo de soporte para el OS finaliza en julio de 2015, el soporte para IIS 6.0 finaliza también en esa fecha.

#### **3.2.4.2 Apache**

Apache es un servidor HTTP de código libre para plataformas UNIX, como BSD y Linux, Windows, Macintosh. Implementa el protocolo HTTP/1.1.

Desarrollado en lenguaje C por la Apache Software Foundation, la primera versión fue lanzada en 1995, y se le considera como el principal valedor del desarrollo inicial de la WWW. Basado inicialmente en el servidor NCSA HTTPd, que posteriormente se eliminó parte del código al ser reescrito, se convirtió en el principal competidor del entonces predominante servidor Netscape Communications Corporation web server hasta alcanzar la actual posición dominante en el mercado.

Apache no fue diseñado con el objetivo de ser el servidor web más rápido del mercado sino para ser altamente configurable y adaptable, y disponer de una gran gama de funcionalidades implementadas en módulos que extienden las capacidades de su núcleo.

#### **3.2.4.3 Nginx**

Nginx es un servidor web y proxy inverso de código libre desarrollado por Igor Sysoev en Nginx, Inc. Su primera versión fue lanzada en 2004, por lo que se considera uno de los servidores web más jóvenes en el mercado.

Soporta los protocolos HTTP, SMTP, POP3 e IMAP y fue diseñado con los objetivos:

- Elevada concurrencia.
- Eficiencia.

- Reducido uso de memoria.

A diferencia de Apache, que utiliza un modelo con un enfoque a procesos y “threads”, Nginx utiliza un modelo enfocado a eventos asíncronos que le proporciona mayor determinismo con elevadas cargas.

Nginx es usado en sitios web como WordPress, Hulu, GitHub, SourceForge y en algunos servicios de Facebook.

Por desgracia, los “builds” de Nginx para Windows se encuentran en estado alfa y por tanto no son indicados para entornos de producción.

#### **3.2.4.4 *Lighttpd***

Lighttpd es un servidor web de código abierto desarrollado por Jan Kneschke. La primera versión fue lanzada en marzo de 2003.

Este servidor web está optimizado para entornos en los que la velocidad es un factor crítico. Implementa características tales como FastCGI y CGI usados como interfaces a programas externos y requiere poca carga de CPU, por lo que lo convierte en un producto ideal para servicios con elevadas peticiones y para distribuir contenido estático.

Aunque inicialmente fue desarrollado como una prueba conceptual de cómo resolver en paralelo 10.000 peticiones en un único servidor, se utiliza de forma generalizada.

Lighttpd es usado en sitios web con un elevado tráfico como Youtube y Wikimedia.

Hasta el momento, no hay ninguna “build” de Lighttpd que se ejecute de forma nativa en Windows. Se requiere para su funcionamiento de Cygwin, un conjunto de herramientas para Windows que proporcionan a las aplicaciones un comportamiento similar a UNIX, por lo que su rendimiento será menor que en entornos UNIX/Linux.

#### **3.2.5 *PriceForecast Web Server***

Tras analizar los productos disponibles en el mercado de servidores web, la única alternativa disponible para Windows Server 2003, descartado IIS por su reducido soporte disponible e imposibilidad de actualización a la versión 7, es Apache Web Server.

Para realizar las pruebas descritas en la sexta sección de este capítulo, se ha desarrollado en Java un servidor web denominado PriceForecast Web Server con el que comparar el rendimiento de Apache Web Server en condiciones de producción.

PriceForecast Web Server se basa en el modelo de “threads” para atender las conexiones entrantes. Implementa el protocolo HTTP/1.0 con únicamente los tres métodos que describe su RFC y no acepta conexiones “keep-alive”; es decir, el servidor tras enviar la respuesta al cliente cierra la conexión. El objetivo de este diseño es evitar que se forme un “pool” de conexiones disponibles en espera de peticiones y así minimizar el uso de los recursos asignados en cada petición y maximizar el ancho de banda disponible para el resto de conexiones. El protocolo HTTP/1.0 está implementado en la totalidad de los navegadores web. Incorpora, además, la posibilidad de establecer conexiones cifradas HTTPS.

Con cada nueva conexión se lanza un nuevo “thread” con el que se atiende la petición del cliente, por lo que la máquina virtual Java (VM) empleará los cuatro núcleos del procesador en el servidor para atender a las peticiones.

PriceForecast Web Server también implementa FastCGI, protocolo de interfaz que permite interconectar programas interactivos con un servidor web. FastCGI es una variante del Common Gateway Interface (CGI) que minimiza la carga y permite más conexiones simultáneas. Proporciona un alto rendimiento. Con ello permitiremos que el servidor procese ficheros PHP de forma eficiente.

La VM de Java contiene un “Security Manager” que junto con una “security policy” define qué recursos están disponibles para la aplicación y antes de ejecutar una instrucción peligrosa, la analiza y posteriormente, la autoriza o deniega en función de su naturaleza. Con ello aumentamos la seguridad en torno al servidor web.

Se consideran seguros en el protocolo HTTP/1.0 los métodos GET y HEAD ya que únicamente solicitan un recurso y no realizan modificaciones en el servidor. El único método inseguro en la versión 1.0 es POST. Al no implementar la versión HTTP/1.1, eliminamos otros métodos inseguros como PUT y DELETE que permiten subir ficheros maliciosos al servidor o eliminarlos.

### 3.3 Sistemas de gestión de bases de datos

#### 3.3.1 Definición

Los sistemas de gestión de bases de datos son un servicio utilizado frecuentemente en los servidores. Permiten almacenar datos siguiendo una determinada lógica y de forma automática en sistemas de información informatizados.

El administrador de bases de datos realizan las tareas de:

- Definir el esquema conceptual de la BD.
- Definir el esquema interno de la BD.
- Vincularse con los usuarios.
- Definir los controles de seguridad y restricciones de integridad.
- Definir los procedimientos de respaldo y recuperación.
- Supervisar el correcto rendimiento y responder a cambios en los requerimientos.

#### 3.3.2 Modelos

Los sistemas de gestión de bases de datos pueden clasificarse en función de su modelo de administración de datos:

- **Modelo jerárquicos:** Almacenan la información en una estructura jerárquica. Con este modelo, los datos se agrupan en forma de árbol. A partir de un nodo denominado padre, cuelgan varios nodos hijos. El nodo sin padre se denomina nodo raíz y los nodos finales sin hijos se denominan hojas. Este esquema se emplea en aplicaciones con elevados volúmenes de información donde se generan estructuras estables.
- **Modelo de red:** Es un modelo similar al jerárquico pero con la diferencia fundamental que permite que un nodo puede tener varios padres, aspecto que no permite el modelo jerárquico.
- **Modelo orientados a objetos:** Modelo reciente en el que se almacenan objetos completos. En este tipo de bases de datos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos.
- **Modelo distribuido:** En este modelo, la base de datos está almacenada en varios ordenadores conectados a la misma red.

- **Modelo documental:** Este modelo organiza la información en documentos identificados con un registro único pero sin un modelo organizativo estricto. Permite realizar búsquedas rápidamente.
- **Modelo relacional:** Es el modelo de gestión de bases de datos más utilizados en la actualidad. Está basado en la lógica de los predicados y en la teoría de conjuntos. Edgar Frank Codd definió sus bases teóricas en 1970 en los laboratorios IBM. El concepto en el que se basan es el uso de relaciones. Todos los datos se almacenan en relaciones, por lo que el lugar y la forma en que se almacenan los datos no tienen relevancia. Para agregar o recuperar la información se realizan consultas en Structured Query Language (SQL). Este lenguaje es un estándar implementado por los principales sistemas de gestión de bases de datos relacionales.

### ***3.3.3 Sistemas de gestión SQL***

Los sistemas de gestión relacionales más utilizados son:

- **MSSQL Server:** Desarrollado por Microsoft únicamente disponible para la plataforma Windows.
- **MySQL:** Es un sistema de gestión de bases de datos multiplataforma, multihilo y multiusuario con licencia GNU GPL. Desarrollada por SUN Microsystems y en la actualidad por Oracle Corporation. Es el sistema de gestión de bases de datos de código abierto más usado en la actualidad. Tiene implementado el motor MyISAM, que desempeña un papel excelente en consultas simples, permite una lectura de datos muy rápida pero provoca errores de integridad en entornos con elevada concurrencia de modificación, por lo que lo hace ideal para aplicaciones de lectura de datos.
- **Oracle:** Es un gestor de base de datos que emplea un modelo objeto-relacional. Está considerado en la actualidad como el sistema más completo y potente pero posee un elevado precio.
- **PostgreSQL:** Gestor de bases de datos de código libre desarrollado en la Universidad de Berkeley en 1982 para entornos de elevada concurrencia y consultas complejas.

### 3.3.4 NoSQL

NoSQL es una amplia gama de sistemas de gestión de bases de datos que difieren claramente del modelo tradicional relacional.

El término NoSQL nació en 1998 para referirse a sistemas de gestión de bases de datos de código abierto, ligeros y que no ofrecen una interfaz SQL pero sí siguen el modelo relacional. Estas aparecieron recientemente cuando las bases de datos relacionales mostraban un rendimiento muy pobre con condiciones de uso de datos intensivos. Ello es debido a que las soluciones típicas RDBMS han sido diseñadas para reducidas transacciones de escritura y lectura, o para un gran conjunto de transacciones pero con pocos accesos de escritura.

La mayoría de las soluciones NoSQL realizan las operaciones en memoria y solo vuelcan los datos al disco duro cada cierto tiempo. Esto permite que las operaciones de escritura sean muy rápidas.

Las características típicas de sistemas NoSQL son:

- **No usan SQL como lenguaje de consultas:** Los gestores NoSQL han sido desarrollados para manejar elevados volúmenes de información que no poseen un esquema fijo.
- **No soportan operaciones JOIN:** Las operaciones JOIN permiten combinar los registros de dos o más tablas. Como en las bases de datos NoSQL la estructura no es fija, estas operaciones no están implementadas.
- **No garantizan completamente ACID:** Son un conjunto de características necesarias que se han de cumplir para que las transacciones sean fiables. Estas características son: atomicidad, en el que cualquier cambio de estado que produce una transacción es atómico; integridad, que asegura que solo se empieza aquello que se puede acabar; aislamiento, propiedad que asegura que no se afectarán entre sí las transacciones; y durabilidad, que asegura la persistencia de una transacción.
- **Escalabilidad Horizontal:** Se realiza para aumentar el rendimiento del sistema simplemente añadiendo más nodos, sin necesidad de ninguna operación a excepción de indicar al sistema cuáles son los nodos disponibles.

Existen disponibles en el mercado diversas soluciones NoSQL como MongoDB, CouchDB, Cassandra, BigTable y HBase.

### 3.3.5 MongoDB

Especial atención merece MongoDB, un sistema multiplataforma de gestión de bases de datos NoSQL potente, flexible y escalable, orientado a documentos de esquema libre. Las características más destacables de MongoDB son su velocidad, ya que está desarrollado en C++, y su rico pero sencillo sistema de consulta de contenidos de la base de datos.

En MongoDB, cada registro o conjunto de datos se denomina documento. Los documentos se agrupan en colecciones, que son una equivalencia a las tablas en una base de datos relacional. La diferencia fundamental es que solo en las colecciones se pueden almacenar documentos de muy distintos formatos, en lugar de estar sometidos a un esquema fijo como en los sistemas tradicionales. MongoDB permite crear índices para algunos atributos de los documentos por lo que mantendrá una estructura interna eficiente para el acceso a la información.

La estructura de cualquier documento es simple y está compuesta por “key-value pairs”, similar a matrices asociativas.

Los documentos se almacenan en formato BSON (Binary JSON), una variación de JSON que permite realizar búsquedas de datos muy rápidas. BSON guarda adicionalmente información útil para el escaneado de datos por lo que ocupa más espacio que si fuese almacenado en formato JSON. Esto es debido a que uno de los principios de los sistemas NoSQL es que el almacenamiento es barato por lo que es mejor aprovecharlo y conseguir un incremento notable de la velocidad de localización de la información.

Un ejemplo de documento MongoDB sería:

```
{
  "_id": ObjectId("8s5d8f125d8f74g58s"),
  "title": "MongoDB",
  "body": "Lorem ipsum...",
  "author_info": {
    "_id": "8s5w8e87t445f877s458",
    "name": "Frank"
  }
}
```

Para interactuar con MongoDB, en su interfaz, se emplea JSON tanto para consultas como para insertar información.

Se considera adecuada la utilización de MongoDB para:

- Almacenamiento y registro de eventos.
- Sistemas de manejo de documentos y contenido.
- Comercio Electrónico.

- Elevados volúmenes de datos.
- Almacén de datos operacional de páginas web.
- Manejo de estadísticas en tiempo real.

## 3.4 Matlab

### 3.4.1 Descripción general

Matlab es un software matemático multiplataforma para cálculo técnico y científico. Incorpora un entorno de desarrollo integrado (IDE) junto con el lenguaje de programación M propio de Matlab creado en 1970.

Desarrollado en 1984 por Cleve Moler en C y Java, Matlab fue adquiriendo prestaciones tales como:

- Manipulación de matrices.
- Representación de datos y funciones.
- Implementación de algoritmos.
- Creación de interfaces de usuario (GUI)
- Comunicación con otros programas escritos en otros lenguajes de programación, como C, C++, Java y Fortran.

Matlab se utiliza en una gran variedad de aplicaciones como las comunicaciones, diseño de control y el procesamiento de señales e imágenes.

### 3.4.2 Matlab Compiler

Desde la versión 5, Matlab incorpora un compilador que permite empaquetar las aplicaciones de Matlab como ejecutables o bibliotecas para su distribución. Éstas utilizan un motor de tiempo de ejecución denominado Matlab Compiler Runtime (MCR).

El compilador Matlab permite ejecutar aplicaciones desarrolladas en Matlab fuera del entorno de Matlab, por lo que se reduce significativamente el periodo de desarrollo de una aplicación ya que no es necesario volver a traducir el código a otro lenguaje de programación.

Al compilar ficheros M, se consigue una mayor velocidad de ejecución, la posibilidad de ocultar algoritmos propietarios, y crear aplicaciones “standalone”.



Para ejecutar la librería o ejecutable compilado en Matlab, el usuario final debe primero instalar el MCR en los equipos donde se ejecutarán. El MCR es completamente compatible con el lenguaje M de Matlab y permite incorporar funciones desde las cajas de herramientas de Matlab.

Matlab incorpora tres posibles formas de compilación: Ficheros MEX, ejecutables en C y C++.

#### **3.4.2.1 Ficheros MEX**

El compilador de Matlab *mcc* traduce el fichero M que contiene la función o aplicación a compilar a un fichero MEX. Los ficheros MEX implementan una interfaz entre Matlab y las subrutinas desarrolladas en C, C++ o Fortran.

Una vez compilado, los ficheros MEX se invocan de forma dinámica y permiten la ejecución de código distinto al lenguaje M.

#### **3.4.2.2 Ejecutables en C**

El compilador Matlab *mcc*, con el “option flag” adecuado (`-e` ó `-m`), convierte ficheros M en código C junto con las librerías:

- Matlab M-File Math Library.
- Matlab Compiler Library.
- Matlab Math Built-In Library.
- Matlab Application Program Interface Library.
- Matlab Utility Library.
- ANSI C Math Library.

#### **3.4.2.3 Ejecutables en C++**

El compilador Matlab *mcc*, con el “option flag” adecuado (`-p`), convierte ficheros M en código C++ junto con las librerías nombradas anteriormente y Matlab C++ Math Library.

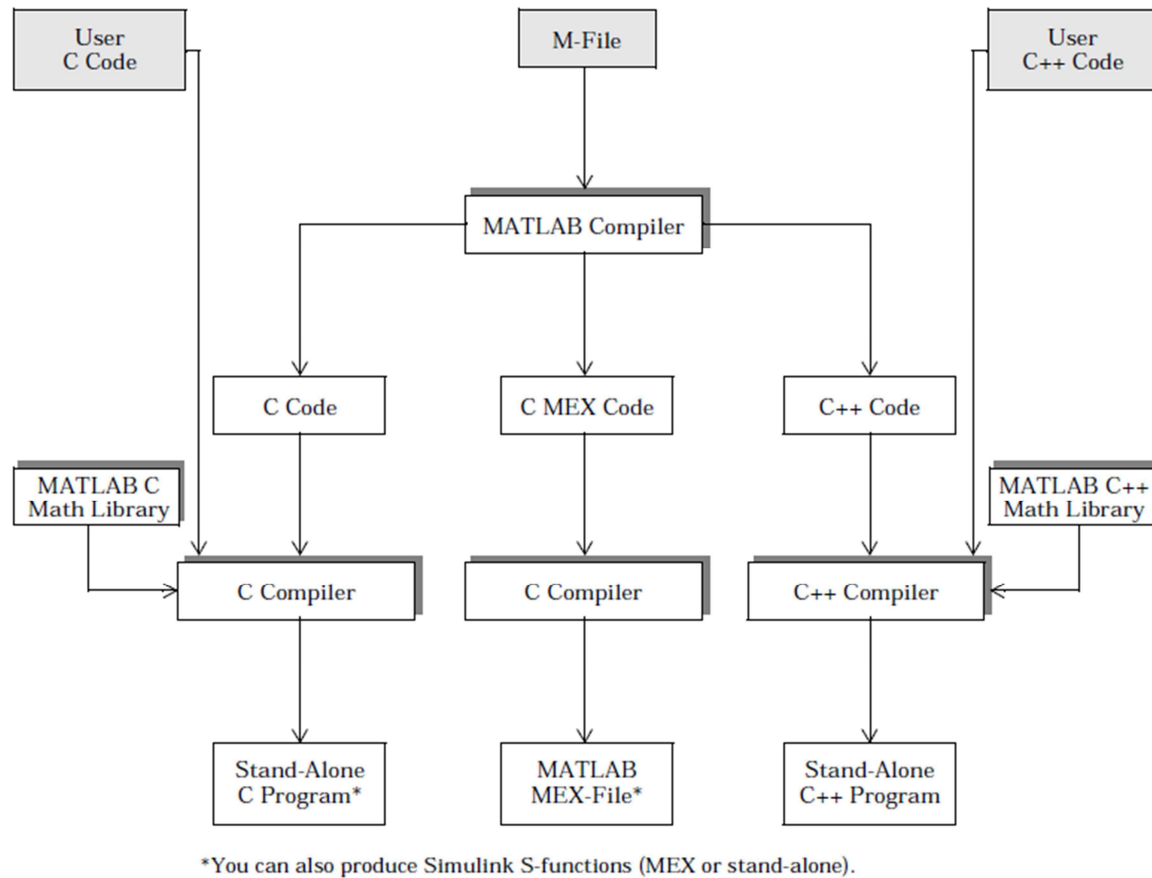


Figura 9. Esquema de compilación en Matlab

Fuente: Matlab Compiler User Guide

### 3.4.3 Matlab PHP-MySQL Gateway

Tradicionalmente, la alimentación de datos a las rutinas desarrolladas en Matlab se ha realizado mediante la lectura de ficheros con un formato de texto plano o “flat file”. Este tipo de ficheros están escritos con caracteres ASCII.

Para ello Matlab viene provisto de comandos como *load*, *textread* o funciones más avanzadas como *fopen* y *fread*. Existe incluso la posibilidad de desarrollar funciones en C para traducir a format *mat* para posteriormente cargarlo con el comando *load*.

Claramente, esta aproximación es insuficiente para casos en los que el volumen de datos a introducir en la rutina es elevado, con distintos formatos y de fuentes variadas. Supone una



sobrecarga sobre el sistema operativo y el disco duro que reduce el rendimiento y velocidad de ejecución.

Afortunadamente, Matlab incorpora una “Database Toolbox” que permite el acceso a cualquier gestor de base de datos con interfaz ODBC/JDBC. Mediante instrucciones en lenguaje SQL, Matlab puede obtener datos de una base de datos conectada con su correspondiente driver.

Sin embargo, la instalación de un driver ODBC no siempre es posible en entornos UNIX y Linux, ni en entornos Windows remotos sin acceso como administrador. Por ello, se desarrolló una pasarela multiplataforma que permite ejecutar sentencias SQL en un gestor de base de datos en donde la solución ODBC no es posible.

Basado en el software ‘Matlab to MySQL Interface’, desarrollado por Luigi Rosa, en la actualidad no funcional y desactualizado, se ha creado una función que permite mediante sockets, ejecutar un fichero PHP en un servidor con acceso al gestor de base de datos. La flexibilidad del lenguaje PHP permite poder tener acceso en Matlab, mediante este software, tanto a bases de datos relacionales como a soluciones de tipo NoSQL.

## 3.5 Lenguajes de programación

### 3.5.1 *Lenguajes compilados e interpretados*

Un lenguaje de programación es un conjunto de instrucciones que forma el código fuente de un programa. Para que un ordenador entienda un programa hay que traducir el código fuente a código máquina. Existen dos maneras posibles de hacerlo:

- El código fuente se traduce a código máquina completamente antes de la ejecución del programada, mediante un proceso llamado compilación. En este caso se dice que el lenguaje de programación es un lenguaje compilado.
- La traducción se realiza en el mismo momento de la ejecución, mediante un proceso de interpretación, en el que un programa llamado intérprete lee las líneas de código, las traduce y las ejecuta. En este caso se dice que el lenguaje de programación es una lenguaje interpretado.

Un lenguaje compilado se ejecuta más deprisa que uno interpretado, pues el compilador ha producido previamente todo el código máquina necesario, y además ha llevado a cabo un proceso de optimización. Como contrapartida, es más fácil desarrollar un intérprete que un compilador.

### 3.5.2 *PHP*

PHP es un lenguaje de programación diseñado y orientado a la creación de páginas web dinámicas. Utilizado generalmente del lado del servidor, actualmente también puede ser utilizado desde una interfaz de línea de comandos o en la creación de aplicaciones con interfaz gráfica.

Es un lenguaje de programación interpretado de código abierto creado originalmente por Rasmus Lerdorf en 1994, aunque actualmente está siendo desarrollado por ‘The PHP Group’ cuyo código sirve como estándar de facto.

Debido a su gran parecido con lenguajes de programación estructurada como C y su corta curva de aprendizaje, en la actualidad se encuentra instado en millones de servidores por todo el mundo. Su principal ventaja reside en ser un lenguaje especialmente preparado para soportar muchos gestores de bases de datos como Oracle, DB2, MySQL, MSSQL Server, e incluso a gestores NoSQL.

### **3.5.3 JavaScript**

JavaScript es un lenguaje basado en objetos, guiado por eventos e interpretado. Desarrollado por Netscape Communications Corporation en 1995, se utiliza principalmente del lado del cliente en aquellos navegadores web que lo tengan implementado.

Permite el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. Diseñado para tener una sintaxis similar a C, ha acabado influenciando a otros lenguajes como ObjectiveJ y JScript.

Este lenguaje puede ser incluido en cualquier documento XHTML y en la actualidad todos los navegadores modernos interpretan código JavaScript. Para interactuar con una página web, el navegador provee de una implementación del Document Object Model (DOM).

JavaScript permite proveer de contenido dinámico a una página web y a la vez, ejecutar instrucciones como respuesta a las acciones de los usuarios.

### **3.5.4 Java**

Java es un lenguaje de alto nivel multiplataforma orientado a objetos. Desarrollado por James Gosling en 1995, incorpora su sintaxis de C, Cobol y Visual Basic pero con un modelo de objetos más sencillo e implementa un recolector de basura.

Java trataba de conseguir tanto la eficiencia de los lenguajes compilados como la universalidad de los lenguajes interpretados.

Los programas escritos en Java sufren una compilación como paso previo a su ejecución. El resultado no es un conjunto de instrucciones comprensibles por un procesador, sino una representación binaria genérica denominada “byte codes”. Se desarrolló una máquina virtual que entendiera los “byte codes”.

La VM se implementa en cada plataforma como software para que funcione como un intérprete de “byte codes”.

Las principales ventajas de Java son:

- Universalidad. Lenguaje Multiplataforma.

Aunque un programa interpretado no es en principio tan rápido como uno compilado, las prestaciones son mucho mejores que las de cualquier lenguaje interpretado.

- Independencia de la plataforma.

Las bibliotecas estándar de funciones y métodos de Java facilitan la programación de multitud de acciones complejas.

- Sencillez.

Gran facilidad de aprendizaje. El código es más legible.

Java dispone de un mecanismo conocido como “recogida de basura”, el cual usando la capacidad multitarea de Java, hace que durante la ejecución de un programa, los objetos que ya no se utilizan se eliminen automáticamente de la memoria. Facilita enormemente el diseño de un programa y optimiza los recursos de la máquina. Con lenguajes tradicionales, la eliminación de objetos y optimización de recursos se debe planificar cuidadosamente.

- Orientado a objetos.

Los lenguajes tradicionales no orientados a objetos como Pascal o C, estaban pensados para trabajar de forma secuencial y basaban su funcionamiento en el concepto de procedimiento y función. La tarea principal del programa se dividía en funciones o tareas más pequeñas, a menudo interrelacionadas, por lo que era difícil modificar una función sin tener que revisar el resto del programa (difícil reutilizar o actualizar programas).

En el caso de lenguajes orientados a objetos, el concepto clave no es la función sino el objeto. Un objeto es un elemento de programación auto contenido y reutilizable. Es la representación en un programa de un concepto y está formado por un conjunto de variables y métodos. La encapsulación de variables y métodos permite que cada objeto sea modificado y mantenido por separado, mantiene en un objeto variables y métodos que no son accesibles desde fuera de él, lo que evita multitud de posibilidades de error (bugs), y permite reutilizar porciones de código ya escrito.

- Seguridad.

La VM posee un verificador de los “byte codes”, que antes de ejecutarlos analiza su formato comprobando que no existen punteros en ellos, que se accede a los recursos del sistema a través de objetos de Java, etc.

## **3.6 XHTML & XML**

### **3.6.1 XHTML**

XHTML, de las siglas “eXtensible HyperText Markup Language”, es un lenguaje utilizado en la maquetación de páginas web. Con él, se define la estructura y contenido de los sitios web mediante etiquetas en texto plano.

La versión 1.0, la utilizada en la actualidad, es el primer tipo de documento de la familia XHTML. Los documentos de la familia XHTML están basados en XML, y son a su vez una evolución del tipo de documento HTML 4.0.

Su finalidad es ser usado como lenguaje de contenidos conforme a XML y, mediante unas sencillas pautas, permitir su funcionamiento en agentes de usuario (por ejemplo, navegadores web) conforme a HTML 4.0. Implementado junto con hojas de estilo CSS ayudan a definir estilos visuales de la web. A su vez, puede incluir scripts que afectan al comportamiento del navegador web y otros procesadores HTML.

XHTML ha supuesto el siguiente paso en la evolución de Internet.

### **3.6.2 XML**

XML, de las siglas “eXtensible Markup Language” es un lenguaje desarrollado por la World Wide Web Consortium a partir del lenguaje SGML, “Standard Generalized Markup Language”, para estructurar documentos.

Se define como un estándar para el intercambio de información estructurada entre diferentes plataformas, da soporte a bases de datos y es útil cuando varias aplicaciones se deben comunicar entre sí o integrar información. Se puede usar en bases de datos, editores de texto, hojas de cálculo, etc.

### 3.7 Librería externas

Este proyecto requiere de librerías en las que apoyarse, todas de código abierto, para poder realizar funciones específicas. Pasamos a continuación a detallar brevemente cada una.

#### 3.7.1 *Twitter Bootstrap*

Twitter Bootstrap es un conjunto de herramientas de “frontend” para el desarrollo rápido de aplicaciones web. Contiene una colección de estilos CSS, HTML y código JavaScript que implementa algunas de las técnicas más modernas de navegación con el mínimo consumo de recursos. Ha sido desarrollado por Mark Otto y Jacob Thornton en Twitter y publicado bajo la licencia Apache.

#### 3.7.2 *jQuery*

jQuery es una librería JavaScript de código abierto desarrollada por John Resig. Consiste en un único fichero JavaScript que contiene las funcionalidades comunes de DOM, evento, efectos y AJAX. El objetivo de esta librería es poder interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos y agregar la técnica AJAX a páginas web. Este tipo de librerías permiten obtener toda una serie de funcionalidades que de otra manera requerirían mucho más código, logrando resultados en menor tiempo.

#### 3.7.3 *pChart*

pChart es una librería desarrollada en PHP por Jean-Damien Pogoletti. Mediante las extensiones GD y FreeType, permite crear una amplia variedad de gráficos a partir de datos obtenidos de ficheros o bases de datos en un servidor. Incorpora un mecanismo denominado pCache para crear gráficos en tiempo real con un menor consumo de CPU y recursos.

#### 3.7.4 *DropBox PHP Lib*

DropBox es un servicio de alojamiento de ficheros en la nube que permite disponer de 2GB gratuitos por usuario. Dispone de herramientas y librerías para los lenguajes de programación



más populares para subir, descargar y manejar los ficheros que se desee. Entre ellas se encuentra una librería para PHP.

### ***3.7.5 Google API PHP Client***

Google liberó un nuevo servicio en abril de 2012 denominado Google Drive. Similar a DropBox pero con mayor espacio gratuito disponible, 5GB. Pone a disposición librerías en varios lenguajes de programación incluido PHP para su manejo. Sin embargo, este servicio todavía no implementa una API completamente funcional.

### ***3.7.6 PHPExcel***

PHPExcel es una librería desarrollada en PHP que permite leer y escribir varios tipos distintos de formato de hojas de cálculo como Excel (BIFF) *xls*, Excel (OfficeOpenXML) *xlsx*, *csv*, Libre/OpenOffice Calc *ods*, etc.

### ***3.7.7 PHP Mailer***

PHP Mailer es una sencilla librería que permite conexiones SMTP e IMAP a servidores de correo electrónico para el envío de emails.

### ***3.7.8 reCAPTCHA***

reCAPTCHA es proyecto patrocinado por Google. Los captchas son un mecanismo en forma de desafío-respuesta utilizada en computación para determinar si el usuario es o no humano. Están diseñados para que no puedan ser comprendidos por una máquina por lo que únicamente un humano podría introducir la secuencia de forma correcta. reCAPTCHA dispone de librería PHP para implementar este mecanismo.

### ***3.7.9 Fcgi4j***

Fcgi4j es una librería para Java de código abierto que implementa el protocolo FastCGI.

### **3.7.10 NicEdit**

NicEdit es un editor de texto desarrollado en JavaScript y AJAX, de código abierto, desarrollado por Brian Kirchoff, que permite editar texto directamente desde el navegador web. Es similar a otros editores tan reconocidos como TinyMCE y posee casi las mismas funcionalidades pero con un tamaño mucho menor, ya que solo ocupa 35 KB.

## **3.8 Web Server Stress Test**

### **3.8.1 ApacheBench (ab)**

ApacheBench es una herramienta mono-thread para medir el rendimiento de los servidores web. Desarrollado conjuntamente con el servidor web Apache para su evaluación, actualmente permite testear el comportamiento de cualquier servidor web bajo unas condiciones de funcionamiento específicas. El objetivo de esta herramienta es la de estimar el rendimiento de un servidor web en entornos con un elevado tráfico.

## **3.9 Web-Based Exploitation**

### **3.9.1 Inyecciones SQL**

Las inyecciones SQL son un método de infiltrar código intruso mediante una vulnerabilidad informática presente en el código fuente de la aplicación a atacar.

La mayoría de las aplicaciones web modernas utilizan lenguajes interpretados junto con bases de datos para almacenar datos y generar contenido dinámico para el usuario. Estas aplicaciones web recogen datos del usuario que se procesan en el servidor.

El origen de este tipo de ataques radica en el incorrecto o nulo chequeo y filtrado de código SQL. El atacante añade código SQL invasor en el código SQL programado inicialmente por lo que se ejecuta todo el código SQL provocando alteraciones en el funcionamiento normal del sistema. El código SQL inyectado puede permitir, en condiciones específicas, insertar, modificar o eliminar registros de una base de datos sin autorización o incluso ejecutar otro tipo de código.

Un ejemplo, en PHP, de este tipo de vulnerabilidad es el siguiente:

```
$sql = "SELECT * FROM users  
      WHERE username = '". $POST['username'] ."'";  
$results = mysql_query($sql);
```

En principio, la variable `$POST['username']` debería contener un nombre como "Alicia" introducido por el usuario en un formulario web.

Pero si incluimos el texto "Alicia' OR 1=1 --", se formará la sentencia SQL:

```
$sql = "SELECT * FROM users WHERE username = 'Alicia' OR 1=1 -- '";
```

SQL considera como comentarios aquel código tras '--', por lo que la sentencia será:

```
$sql = "SELECT * FROM users WHERE username = 'Alicia' OR 1=1";
```

Como la sección de código `OR 1 = 1` siempre se cumple, la sentencia equivalente final será:

```
$sql = "SELECT * FROM users";
```

Con esta sentencia, el gestor de bases de datos dará como resultado todos los datos contenidos en la tabla `users` que pueden acabar mostrándose en la página web.

La mayoría de los lenguajes de programación enfocados a la web disponen de herramientas que evitan este tipo de ataques clásicos que filtran y verifican las consultas que se van a realizar.

### ***3.9.2 Inyecciones NoSQL***

Debido a que los gestores de bases de datos NoSQL no utilizan lenguaje SQL, a priori no son vulnerables a este tipo de ataques.

Por desgracia, los motores de bases de datos NoSQL que procesen código JavaScript también pueden ser vulnerables a este tipo de ataques. MongoDB soporta el uso de funciones javascript en las "query" por lo que puede ser vulnerable.

Este vector de ataques web es reciente, apenas documentado y no existen herramientas de escaneado automático disponibles en la actualidad.

Para evitar estos ataques es conveniente evitar generar comandos de JavaScript "ad-hoc" mediante la concatenación de scripts con datos proporcionados por el usuario y comprobar que los datos son filtrados antes de ser enviado al gestor de bases de datos.

### **3.9.3 Cross-Site Scripting (XSS)**

El XSS es un proceso de inyección de código script en la aplicación web cuyo objetivo no es el ataque de un servidor sino de los clientes. El código inyectado puede residir o bien en la página web original o puede ser procesado por el navegador web de cada cliente, ejecutándose como si fuera parte del código original.

Como medida de seguridad, las aplicaciones web están diseñadas para tener acceso únicamente a los datos de sí misma en cada cliente conectado. Esto implica, que la información almacenada de una aplicación web en un cliente no puede ser accesible por otra aplicación web distinta. Debido a que el código malicioso se ejecuta en el entorno de la aplicación web objeto de ataque, éste tiene acceso a información potencialmente sensible almacenada en el navegador web del cliente, incluyendo “tokens” de sesión y “cookies”.

### **3.9.4 Nikto**

Nikto es un scanner de vulnerabilidades en servidores web. Desarrollado en lenguaje Perl por Chris Sullo y David Logde, este software permite automatizar el proceso de escaneo de servidores web desactualizados y no parcheados así como la búsqueda de archivos potencialmente peligrosos que pueden residir en el servidor. Nikto es capaz de identificar un amplio rango de cuestiones específicas y comprueba la configuración en servidores web.

## **3.10 Port & Vulnerability Scanning**

Una red informática está constituida por un conjunto de herramientas que permiten a los ordenadores que la integran compartir información y recursos. Cada ordenador que forma la red se denomina nodo y para comunicarse entre sí, los nodos utilizan protocolos.

La mayoría de las redes deben, por tanto, permitir que fluya información en ella. Las redes, en total aislamiento sin conexión a Internet, sin servicios como el correo electrónico o el tráfico de Internet, son muy raras en la actualidad. Cada servicio y conexión a otra red ofrece un vector de ataque potencial.

Los puertos proporcionan un canal para que tanto el software como las redes puedan comunicarse con el hardware. Un puerto es una conexión de datos que permite a un ordenador intercambiar información con otros ordenadores, software o dispositivos. Hay un total de 65.536

(0 - 65.535) puertos en cada ordenador. Los puertos pueden ser TCP o UDP dependiendo del servicio que utiliza el puerto o la naturaleza de comunicación.

Muchos servicios en red se ejecutan en puertos estándar que puede dar a los atacantes una indicación de la función del sistema bajo ataque. Por ejemplo, un servidor con el puerto 80 abierto, se le puede intentar abrir una conexión a ese puerto y con frecuencia, se podrá obtener información específica sobre el servidor web que está escuchando en ese puerto específico.

### **3.10.1 Nmap**

Nmap es una herramienta de código abierto desarrollada por Gordon “Fyodor” Lyon. Su objetivo es efectuar un rastreo de puertos con el fin de evaluar la seguridad de sistemas informáticos.

Entre los métodos de identificación de servicios que implementa destacan:

- **TCP Connect Scan**

Es el escaneado de puertos más estable y básico debido a que completa la secuencia “Three-Way Handshake”. Esta secuencia se inicia con el envío del cliente al servidor de un paquete SYN a un puerto específico. Si el servidor tiene el puerto abierto, responde al cliente con un paquete SYN/ACK. En cuanto el cliente recibe este último paquete, responde con un paquete ACK. A partir de este momento la comunicación entre ambas máquinas queda establecida.

- **TCP SYN Scan**

Usando este método, el cliente, al recibir el paquete SYN/ACK durante el “Three-Way Handshake”, responde con un paquete RST en vez del paquete ACK lo que provoca que la comunicación no quede establecida entre las máquinas. En este caso, al no establecerse comunicación en ningún momento, el escaneado pasa más desapercibido que en la modalidad anterior.

- **UDP Scan**

El protocolo TCP se considera orientado a la conexión ya que garantiza que los paquetes enviados lleguen al receptor de forma intacta y en el orden específico. El protocolo UDP, por otro lado, sencillamente envía los paquetes al receptor sin ningún mecanismo que compruebe que han llegado de forma correcta.

Con este escaneado, la herramienta Nmap envía no solo paquetes al puerto especificado sino paquetes específicos para comprobar que los puertos posiblemente abiertos lo están.

- **Xmas Scan**

En este tipo de escaneado de puertos, la herramienta envía un paquete con los “flags” FIN, PSH y URG activados mientras que los “flags” SYN y ACK están desactivados.

La RFC del protocolo TCP especifica que si un puerto cerrado recibe este tipo de paquete, la máquina debe responder con un paquete RST. En cambio, si el puerto está abierto, este tipo de paquetes se ignora. Esto permite a Nmap identificar si un puerto está realmente cerrado.

### ***3.10.2 Nessus***

Nessus es una aplicación para el escaneo de vulnerabilidades en diversos S.O. Consta de un “demonio” que realiza el escaneo del servidor objetivo y de un cliente con una interfaz web que permite al usuario interactuar con la herramienta y visualizar los resultados. A diferencia de nmap, Nessus incluye una larga lista de plugins escritos en su propio lenguaje script ‘Nessus Attack Scripting Language’ para realizar pruebas de vulnerabilidad. Nessus fue desarrollado en 1998 por Renaud Deraison.

## **3.11 Server Exploitation**

El proceso de “exploit” consiste en obtener control de un sistema informático. Éste puede realizarse de muchas maneras y hay disponibles multitud de herramientas para ello. El objetivo último en cualquier caso es obtener un acceso de tipo administrador en la máquina bajo ataque.

Los “exploits” son piezas de software o secuencia de comandos con el objetivo de causar un error en la máquina víctima y causar un comportamiento no deseado o imprevisto por los desarrolladores. El fin último es violar las medidas de seguridad y acceder de forma no autorizada.

### ***3.11.1 Metasploit Framework***

Metasploit es un software destinado a la seguridad informática, de código abierto, que incluye un “exploit framework”, una estructura que permite desarrollar y lanzar “exploits”. Asiste al usuario en el proceso de desarrollo.

Esta herramienta contiene una amplia variedad de “payloads”. Los “payloads” son piezas de código que se ejecutarán en la máquina deseada una vez el usuario tiene acceso a la misma con el objetivo de realizar funciones adicionales como añadir usuarios, abrir puertas traseras o instalar software que permita mantener el acceso aún en el caso de que el servidor se reinicie.

### ***3.11.2 Fast-Track Autopwn***

Fast-Track Autopwn es una herramienta similar a Metasploit con la única diferencia de que automatiza todo el proceso, ya que lo único que requiere es una dirección IP.

Esta herramienta realiza en primer lugar un escaneo de puertos como Nmap. Una vez listado los puertos abiertos lanza todos los “exploits” de los que dispone contra la máquina víctima. Aun en el caso de que Fast-Track haya accedido en un momento dado del proceso a una línea de comandos lista, seguirá lanzando el resto de exploits hasta finalizar. En muchas ocasiones, se obtiene más de un acceso a la máquina como resultado.

## **3.12 Cortafuegos & antivirus**

Un cortafuegos es un sistema de seguridad cuya función es filtrar las comunicaciones entre redes, facilitando únicamente las autorizadas y evitando accesos ilícitos. Recoge todo el tráfico que circula por la red que monitoriza.

Entre sus principales objetivos destacan: evitar que usuarios sin autorización accedan a redes privadas e implementar políticas de seguridad específicas.

Existen dos tipos de cortafuegos:

- **Cortafuegos por software:** Se instala un programa en la máquina a proteger que actúa como cortafuegos. Su precio suele ser reducido y hay incluso soluciones gratuitas.

- **Cortafuegos por hardware:** Son más complejos y suponen un mayor coste. Estos cortafuegos suelen estar instalados en routers con varios ordenadores conectados a él.

Al instalar un cortafuegos, se logra una mayor seguridad frente a ataques ilícitos desde el exterior, pero no protegen completamente el sistema de posible ataques como los internos.

Un antivirus es un software de seguridad que debe ser instalado en la máquina, cuyo objetivo es detectar y eliminar virus informáticos, así como otro tipo de “malware” como los rootkits.



## Capítulo 4

### Análisis, diseño e implantación del Backend

#### 4.1 Análisis

En esta fase del proyecto, se presenta una visión general del proceso de análisis como paso previo a la implementación de la lógica del sistema.

##### 4.1.1 Requisitos funcionales

Los requisitos funcionales (RF) de un sistema especifican las acciones que debe realizar y la información que el sistema debe contener.

RF-01			
Necesidad	Esencial	Fuente	Tutores
Descripción			
El sistema descargará cada día los datos de precios horarios de casación de la web de OMIE.			

Tabla 2. RF-01

RF-02			
Necesidad	Esencial	Fuente	Tutores
Descripción			
El sistema descargará cada día los datos de previsión de demanda horaria de la web de REE.			

Tabla 3. RF-02

RF-03			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema descargará a principios de cada mes los datos de previsión de demanda mensual de la web de REE.			

Tabla 4. RF-03

RF-04			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema interpolará los datos de demanda horaria no disponibles.			

Tabla 5. RF-04

RF-05			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema generará predicciones de precios horarios, mensuales y pico.			

Tabla 6. RF-05

RF-06			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema calculará los errores de predicción acumulados.			

Tabla 7. RF-06

RF-07			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema almacenará todos los datos requeridos para las predicciones así como los resultados obtenidos.			

Tabla 8. RF-07

RF-08			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema generará una gráfica de cada predicción de precios realizada.			

Tabla 9. RF-08

RF-09			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema generará gráficas de errores acumulados.			

Tabla 10. RF-09

RF-10			
<b>Necesidad</b>	Deseable	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema generará gráficas de los datos de entrada relevantes para el usuario.			

Tabla 11. RF-10

#### 4.1.2 Requisitos no funcionales

Los requisitos no funcionales (RFN) se refieren a características del sistema relacionados con el rendimiento, la plataforma y aquellas condiciones que delimitan los requisitos funcionales.

RFN-01			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema debe ejecutarse en Windows Server 2003.			

Tabla 12. RFN-01

RFN-02			
<b>Necesidad</b>	Deseable	<b>Fuente</b>	Desarrollador
<b>Descripción</b>			
El sistema debe ser portable y ejecutable en cualquier SO moderno.			

Tabla 13. RFN-02

RFN-03			
<b>Necesidad</b>	Deseable	<b>Fuente</b>	Desarrollador
<b>Descripción</b>			
El sistema estará programado con tecnologías de código abierto.			

Tabla 14. RFN-03

RFN-04			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema debe garantizar la integridad y disponibilidad de los datos en todo momento.			

Tabla 15. RFN-04

RFN-05			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema debe tener implementado un sistema para generar copias de seguridad de los datos y poder restaurarlos en caso de necesidad.			

Tabla 16. RFN-05

RFN-06			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema debe ser estable en condiciones normales.			

Tabla 17. RFN-06

RFN-07			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema generará gráficas para cada predicción de precios generada que se almacenarán de forma permanente en el servidor.			

Tabla 18. RFN-07

RFN-08			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Desarrollador
<b>Descripción</b>			
En caso de caída prolongada del servidor, el sistema deberá actualizar los datos y generar todas las predicciones pendientes hasta la fecha actual.			

Tabla 19. RFN-08

RFN-09			
<b>Necesidad</b>	Esencial	<b>Fuente</b>	Tutores
<b>Descripción</b>			
El sistema debe ser seguro frente a ataques informáticos.			

Tabla 20. RFN-09

## 4.2 Diseño

### 4.2.1 Modelo conceptual

El diseño del “backend” del sistema se muestra a continuación mediante un diagrama UML de actividad que especifica las fases entre el inicio y el final de la ejecución.

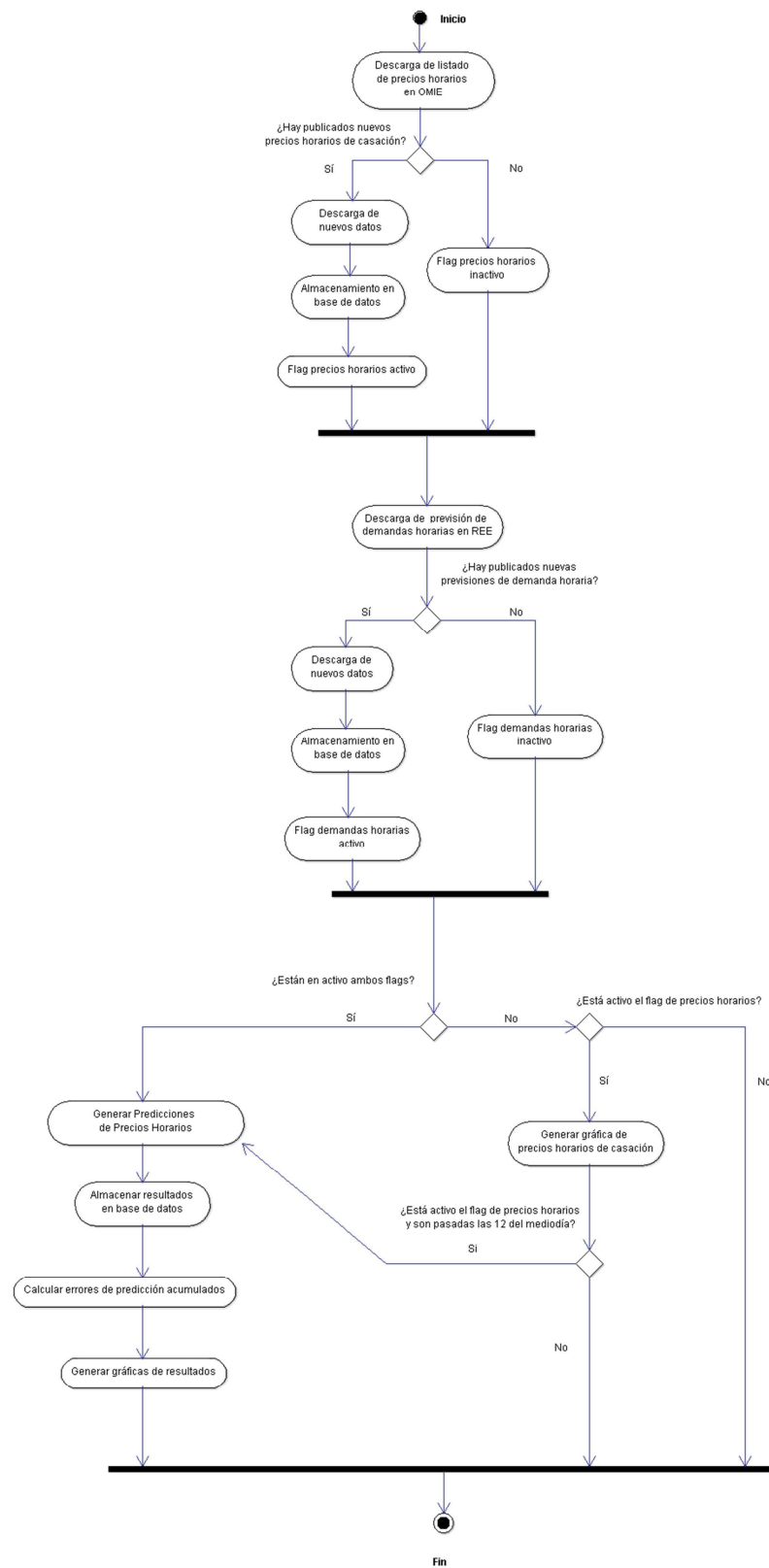


Figura 10. Diagrama UML de actividades para predicciones horarias

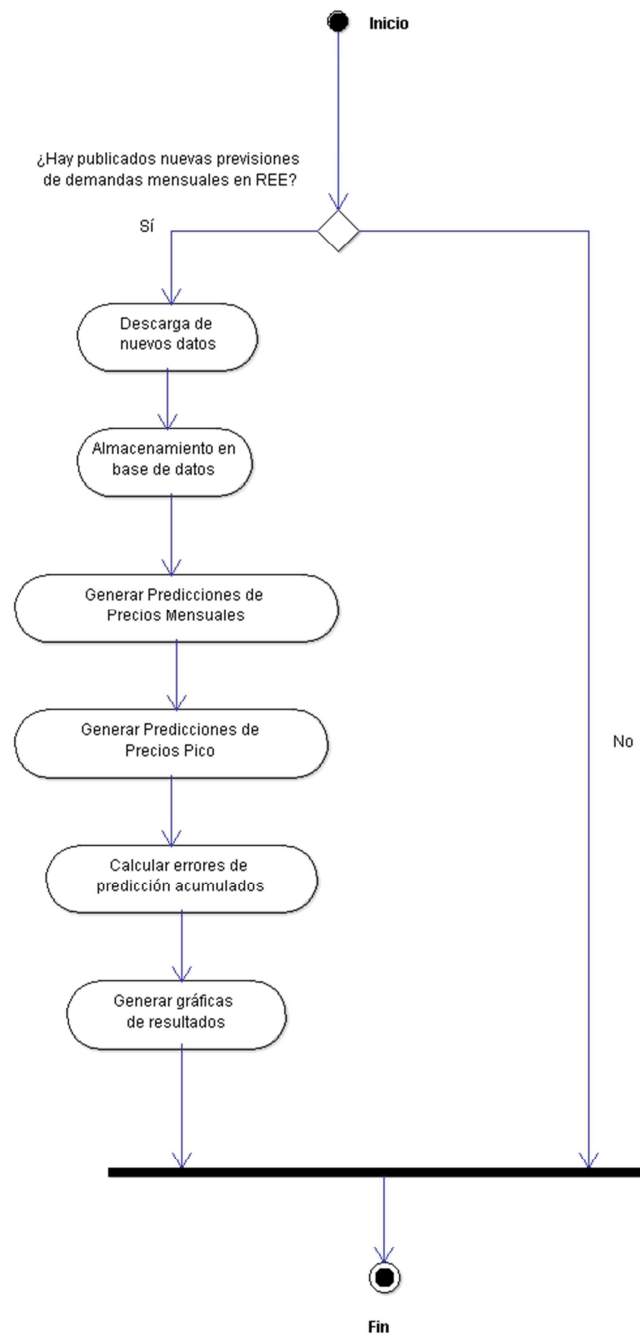


Figura 11. Diagrama UML de actividades para predicciones mensuales y de pico

#### 4.2.2 Modelo de la base de datos

La base de datos ‘priceforecasting’ en el servidor MySQL únicamente está destinado a almacenar de forma permanente los datos de OMIE y REE y los resultados de las predicciones obtenidas por el sistema; así como los datos adicionales que el sistema calcula internamente.

Procedemos a continuación a mostrar la estructura de la base de datos.

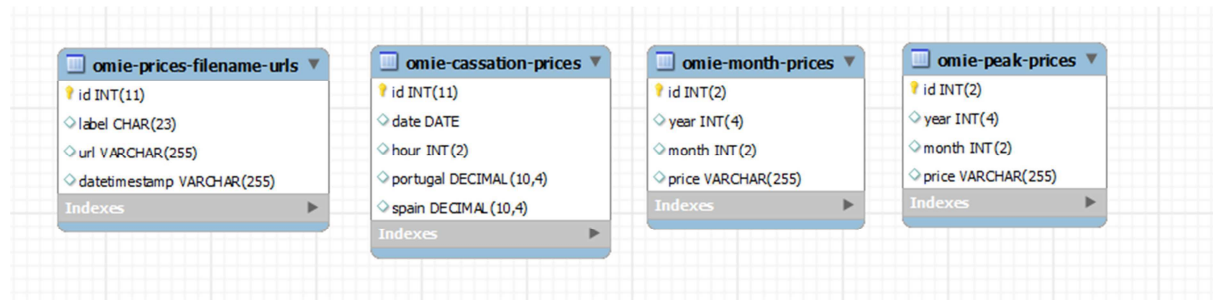


Figura 12. Modelo de base de datos I

- **omie-prices-filename-urls:** Almacena los ficheros de datos de precios horarios de casación publicados por OMIE.
- **omie-cassation-prices:** Almacena los precios horarios de casación publicados por OMIE.
- **omie-month-prices:** Almacena los precios mensuales calculados por el sistema como media de todos los precios horarios del mes.
- **omie-peak-prices:** Almacena los precios pico calculados por el sistema como media de los precios horarios entre las 8:00h y las 20:00h de cada día del mes.

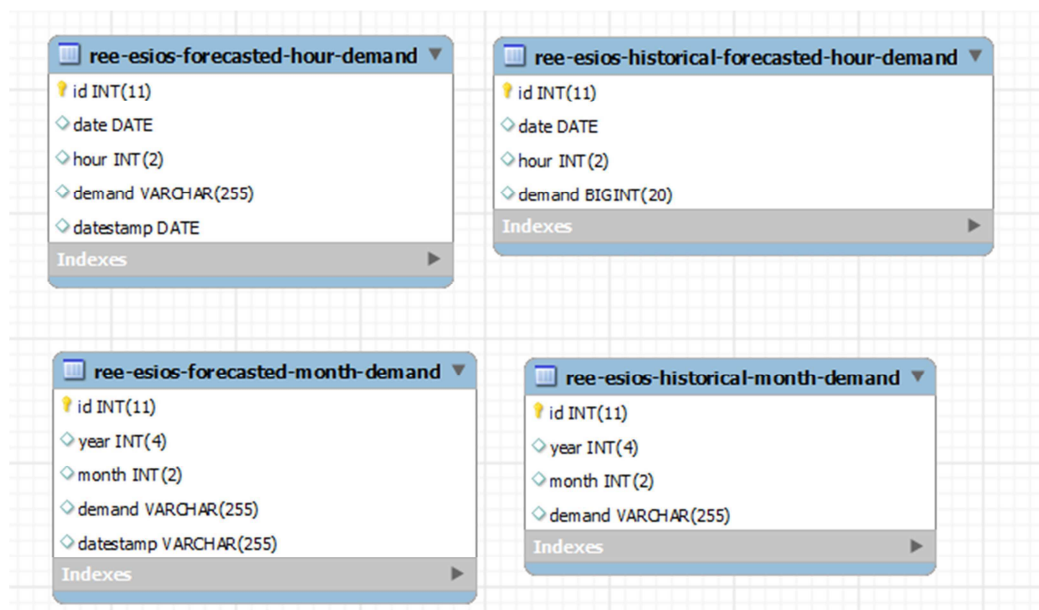


Figura 13. Modelo de base de datos II



- **ree-esios-forecasted-hour-demand:** Almacena las previsiones horarias de demanda publicadas cada día por REE.
- **ree-esios-historical-forecasted-hour-demand:** Almacena las previsiones horarias finales de demanda.
- **ree-esios-forecasted-month-demand:** Almacena las previsiones mensuales de demanda publicadas cada mes por REE.
- **ree-esios-historical-month-demand:** Almacena las previsiones mensuales finales de demanda.

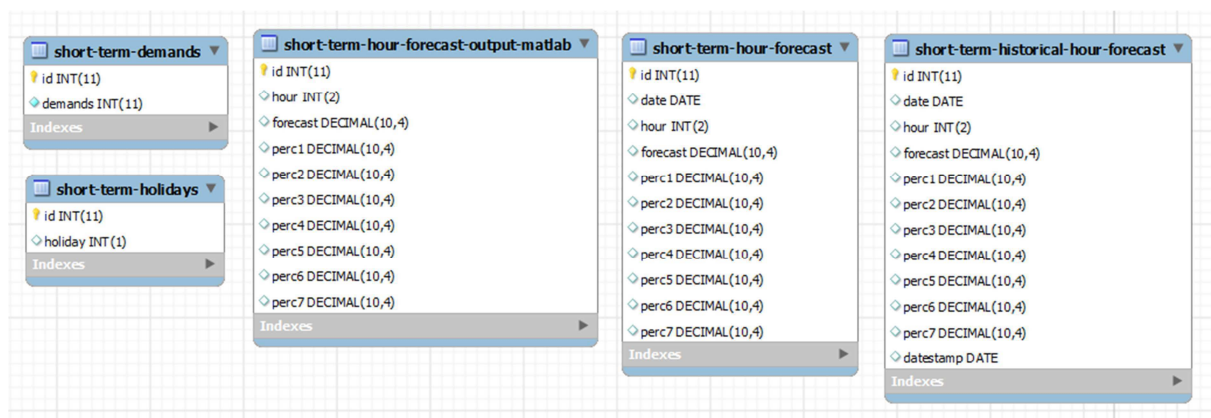


Figura 14. Modelo de base de datos III

- **short-term-demands:** Almacena temporalmente los datos de demanda que requiere la rutina Matlab de predicciones horarias.
- **short-term-holidays:** Almacena temporalmente los datos de festivos que la rutina requiere.
- **short-term-hour-forecast-output-matlab:** Almacena temporalmente el volcado en bruto de la rutina Matlab.
- **short-term-hour-forecast:** Almacena temporalmente los datos de predicción procesados.
- **short-term-historical-hour-forecast:** Almacena las predicciones horarias realizadas.

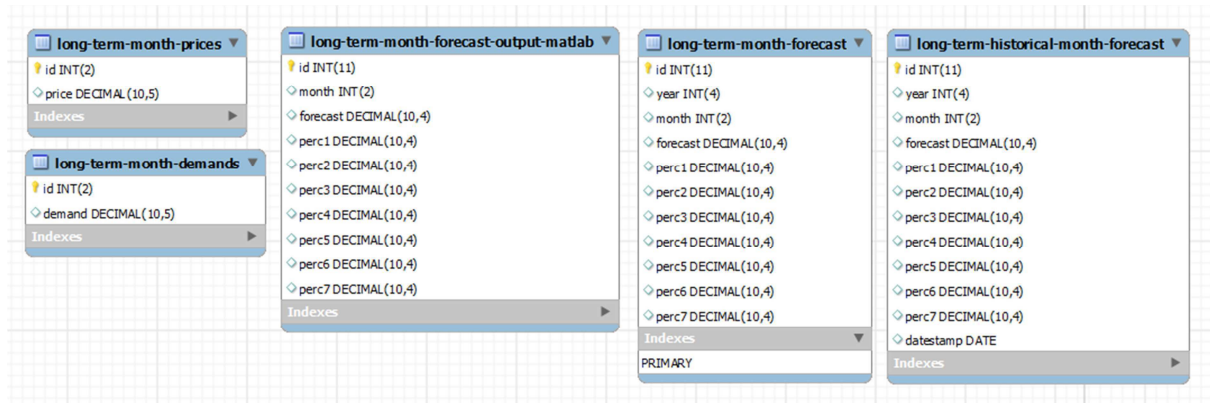


Figura 15. Modelo de la base de datos IV

- **long-term-prices:** Almacena temporalmente los datos de precio que requiere la rutina Matlab de predicciones mensuales.
- **long-term-demands:** Almacena temporalmente los datos de demanda que requiere la rutina Matlab de predicciones mensuales.
- **long-term-month-forecast-output-matlab:** Almacena temporalmente el volcado en bruto de la rutina Matlab.
- **long-term-month-forecast:** Almacena temporalmente los datos de predicción procesados.
- **long-term-historical-month-forecast:** Almacena las predicciones mensuales realizadas.

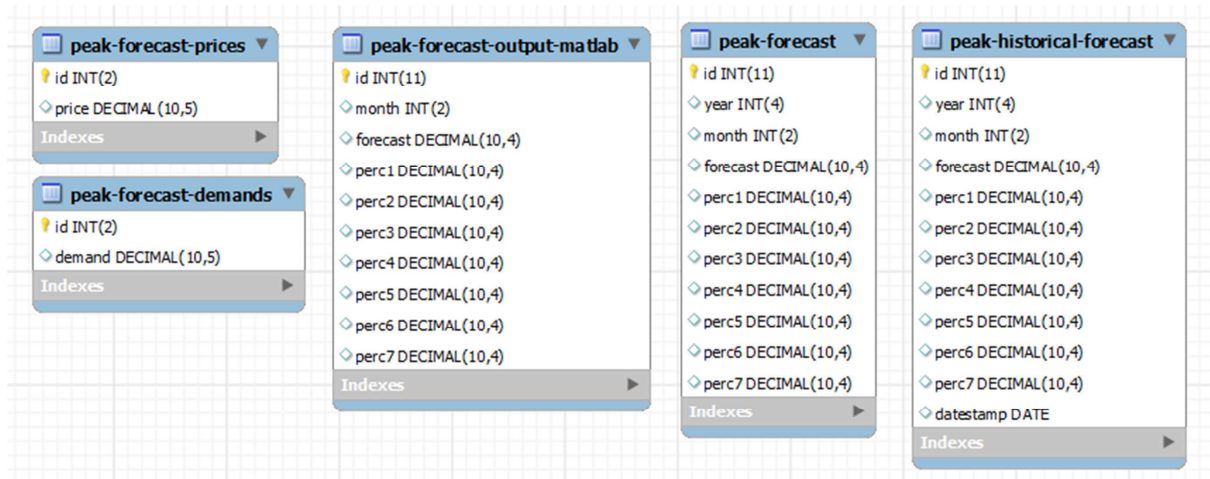


Figura 16. Modelo de base de datos V

- **peak-forecast-prices:** Almacena temporalmente los datos de precio que requiere la rutina Matlab de predicciones de pico.
- **peak-forecast-demands:** Almacena temporalmente los datos de demanda que requiere la rutina Matlab de predicciones de pico.
- **peak-forecast-output-matlab:** Almacena temporalmente el volcado en bruto de la rutina Matlab.
- **peak-forecast:** Almacena temporalmente los datos de predicción procesados.
- **peak-historical-forecast:** Almacena las predicciones de pico realizadas.

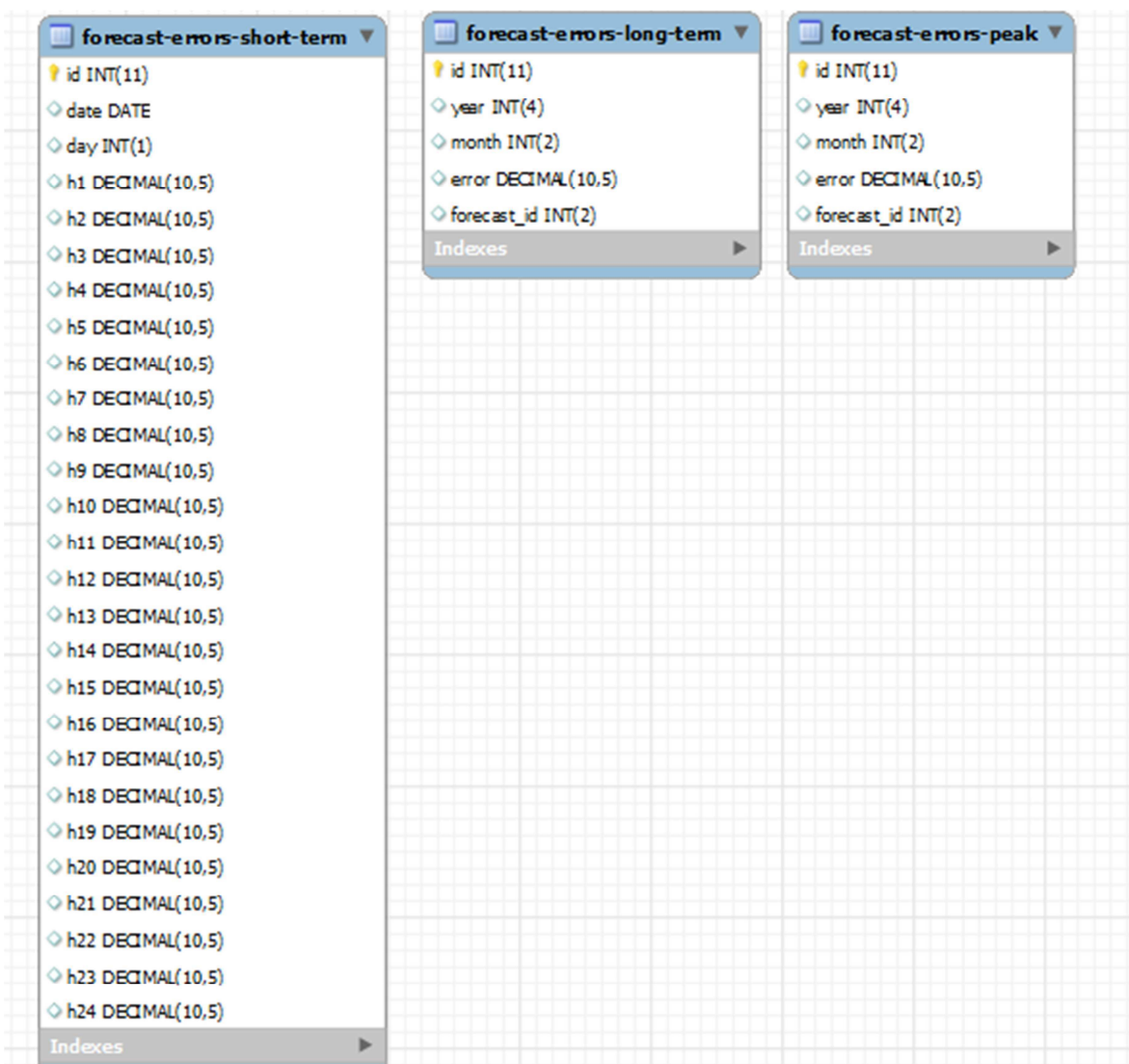


Figura 17. Modelo de la base de datos VI

- **forecast-errors-short-term:** Almacena los errores acumulados de predicción de los precios horarios.
- **forecast-errors-long-term:** Almacena los errores acumulados de predicción de precios mensuales.
- **forecast-errors-peak:** Almacena los errores acumulados de predicción de precios pico.

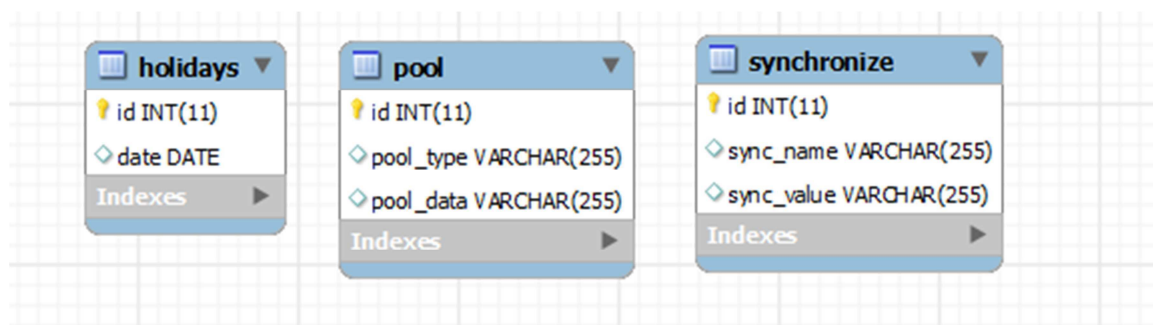


Figura 18. Modelo de base de datos VII

- **holidays:** Almacena las fechas de festivos oficiales.
- **pool:** Almacena las acciones indicadas en la aplicación web.
- **synchronize:** Almacena los flags de uso interno por el sistema.

## 4.3 Implementación

Una vez definidos los requisitos del sistema, tanto funcionales como no funcionales, se procede con la fase de desarrollo e implementación.

### 4.3.1 Adquisición de datos

Para la predicción de precios de la energía eléctrica para los horizontes horario, mensual y pico, el sistema requiere tres fuentes de datos:

- **Precios horarios de casación**

Se obtienen en la web pública de OMIE, en la URL

<http://www.omie.es/aplicaciones/datosftp/datosftp.jsp?path=/marginalpdbc/>

En esta página web aparece un listado de todos los precios de casación horarios disponibles, organizados en documentos, con un nombre de fichero estructurado de la forma `marginalpdbc_aaaamdd.1`

Cada fichero tiene la siguiente estructura:

```
MARGINALPDBC;  
AÑO;MES;DIA;HORA;PRECIO EN PORTUGAL;PRECIO EN ESPAÑA;  
*
```

Ejemplo:

```
MARGINALPDBC;  
2012;09;02;1;40.19;40.19;  
2012;09;02;2;34.27;34.27;  
2012;09;02;3;27.06;25;  
...  
2012;09;02;23;52;52;  
2012;09;02;24;47;47;  
*
```

Cabe señalar que desde el 1 de enero de 2010, se produjo un cambio en el formato de datos. Hasta entonces, los precios para España y Portugal eran en céntimos de €/KWh. Desde el 1 de enero de 2010, son en €/MWh.

- **Demandas horarias de energía eléctrica prevista por REE**

Se obtienen del sistema de información público ESIOS, a través de una URL estructurada de la forma:

`http://www.esios.ree.es/Solicitar/demanda_aux_AAAAMMDD.xml`

Los ficheros que se obtienen están en formato xml con la siguiente estructura:

```
<Demanda>
  <IdentificacionMensaje v="demanda_AAMMDD"/>
  <VersionMensaje v="1"/>
  <TipoMensaje v="Z01"/>
  <TipoProceso v="A01"/>
  <TipoClasificacion v="A02"/>
  <IdentificacionRemitente v="10XES-REE-----E"
codificacion="A01"/>
  <FuncionRemitente v="A04"/>
  <IdentificacionDestinatario v="CODIGOSUJETO"
codificacion="A01"/>
  <FuncionDestinatario v="A08"/>
  <FechaHoraMensaje v="..."/>
  <Horizonte v="..."/>
  <SeriesTemporales>
    <Periodo>
      <IntervaloTiempo v="..."/>
      <Intervalo>
        <Pos v="HORA"/>
        <Ctd v="DEMANDA"/>
      </Intervalo>
    </Periodo>
  </SeriesTemporales>
</Demanda>
```

Ejemplo:

```
<Demanda>
  <IdentificacionMensaje v="demanda_20120901"/>
  <VersionMensaje v="1"/>
  <TipoMensaje v="Z01"/>
  <TipoProceso v="A01"/>
  <TipoClasificacion v="A02"/>
  <IdentificacionRemitente v="10XES-REE-----E"
codificacion="A01"/>
  <FuncionRemitente v="A04"/>
  <IdentificacionDestinatario v="CODIGOSUJETO"
codificacion="A01"/>
  <FuncionDestinatario v="A08"/>
  <FechaHoraMensaje v="2012-08-25T04:49:45Z"/>
  <Horizonte v="2012-08-31T22:00Z/2012-09-01T22:00Z"/>
  <SeriesTemporales>
    <Periodo>
      <IntervaloTiempo v="2012-08-31T22:00Z/2012-09-
01T22:00Z"/>
      <Intervalo>
```

```
<Pos v="1"/>
<Ctd v="26777"/>
</Intervalo>
<Intervalo>
  <Pos v="2"/>
  <Ctd v="25162"/>
</Intervalo>
...
<Intervalo>
  <Pos v="23"/>
  <Ctd v="28469"/>
</Intervalo>
<Intervalo>
  <Pos v="24"/>
  <Ctd v="26547"/>
</Intervalo>
</Periodo>
</SeriesTemporales>
</Demanda>
```

Los datos de demandas horarias son en MWh.

- **Demandas mensuales de energía eléctrica prevista por REE**

Se obtienen del sistema de información público ESIOS, a través de una URL estructurada de la forma:

[http://www.esios.ree.es/Solicitar/pubpredemmensual\\_AAAAMM.xml](http://www.esios.ree.es/Solicitar/pubpredemmensual_AAAAMM.xml)

Los ficheros que se obtienen están en formato xml con la siguiente estructura:

```
<PubPreDemMensual
xmlns="http://sujetos.esios.ree.es/schemas/2007/04/30/PubPreDem
Mensual-esios-AplicMP/">
  <IdentificacionMensaje v="pubpredemmensual_AAAAMM"/>
  <VersionMensaje v="2"/>
  <TipoMensaje v="Z18"/>
  <TipoProceso v="Z41"/>
  <TipoClasificacion v="Z45"/>
  <IdentificacionRemitente codificacion="A01" v="..."/>
  <FuncionRemitente v="A04"/>
  <IdentificacionDestinatario codificacion="A01" v="..."/>
  <FuncionDestinatario v="A04"/>
  <FechaHoraMensaje v="..."/>
  <Horizonte v="..."/>
  <SeriesTemporales>
    <IdentificacionSeriesTemporales v="STP0"/>
    <UnidadMedida v="GWH"/>
    <Periodo>
      <IntervaloTiempo v="..."/>
      <Resolucion v="P01M"/>
      <Intervalo>
        <Pos v="MES"/>
        <Ctd v="DEMANDA"/>
      </Intervalo>
```

```
        </Periodo>
    </SeriesTemporales>
</PubPreDemMensual>
```

Ejemplo:

```
<PubPreDemMensual
xmlns="http://sujetos.esios.ree.es/schemas/2007/04/30/PubPreDem
Mensual-esios-AplicMP/">
    <IdentificacionMensaje v="pubpredemmensual_201208"/>
    <VersionMensaje v="2"/>
    <TipoMensaje v="Z18"/>
    <TipoProceso v="Z41"/>
    <TipoClasificacion v="Z45"/>
    <IdentificacionRemitente
        codificacion="A01" v="10XES-REE-----E"/>
    <FuncionRemitente v="A04"/>
    <IdentificacionDestinatario
        codificacion="A01" v="10XES-REE-----E"/>
    <FuncionDestinatario v="A04"/>
    <FechaHoraMensaje v="2012-08-03T11:49:20Z"/>
    <Horizonte v="2012-08-31T22:00Z/2013-09-01T22:00Z"/>
    <SeriesTemporales>
        <IdentificacionSeriesTemporales v="STP0"/>
        <UnidadMedida v="GWH"/>
        <Periodo>
            <IntervaloTiempo
                v="2012-08-31T22:00Z/2013-09-01T22:00Z"/>
            <Resolucion v="P01M"/>
            <Intervalo>
                <Pos v="1"/>
                <Ctd v="19418"/>
            </Intervalo>
            ...
            <Intervalo>
                <Pos v="12"/>
                <Ctd v="21051"/>
            </Intervalo>
        </Periodo>
    </SeriesTemporales>
</PubPreDemMensual>
```

Los datos de demandas mensuales son en GWh.

#### 4.3.2 Procesado de datos

Los datos de precios horarios de casación publicados por OMIE, y las predicciones de demanda horaria y mensuales publicadas por REE, se almacenan de forma permanente en una base de datos en el servidor MySQL denominada 'priceforecasting'. En el apartado 4.2.2. aparecen detallados la estructura y el modelo de la base de datos.



El procesado de datos se basa en la activación y desactivación de flags. Los flags son señalizaciones almacenadas en la base de datos que permiten controlar o si están disponibles todos los datos que las rutinas Matlab de predicción de precios requieren y evitar generar datos erróneos.

#### ***4.3.3 Interpolación de datos***

Uno de los datos que requiere la rutina de predicción de precios horarios es la previsión de demanda eléctrica para los próximos 7 días. Estos datos los elabora REE, que dispone de sus propias herramientas para realizar la previsión. Aproximadamente a las 6:00h de la mañana, los datos quedan disponibles cada día para los próximos 7 días.

Ocasionalmente, REE considera por factores diversos, que sus herramientas de predicción de demanda no son adecuadas para generar la predicción en días señalados como atípicos. Un ejemplo de ello es aquellos años en los que el día 2 de mayo, festivo en la Comunidad de Madrid cae un domingo, y la Comunidad decide trasladar el festivo al lunes siguiente. En estos casos, la demanda horaria para ese día concreto no aparece disponible hasta el mismo día de madrugada.

La rutina implementada en este proyecto requiere de todos los datos de previsión de demanda para poder realizar predicciones adecuadas.

Por ello, se consideró, inicialmente, que lo más indicado era usar un modelo sencillo como el alisado exponencial con tendencia y estacionalidad, es decir, como el modelo ‘Holt-Winter’. Este modelo exige un factor de tendencia y otro de estacionalidad para realizar las predicciones.

Para confirmar si los datos presentan tendencia y estacionalidad, se realizó una gráfica con los datos de demandas horarias desde el 1 de enero de 2010, puestas a disposición en el sistema ESIOs de REE, tomando como referencia la hora 12 de cada día.

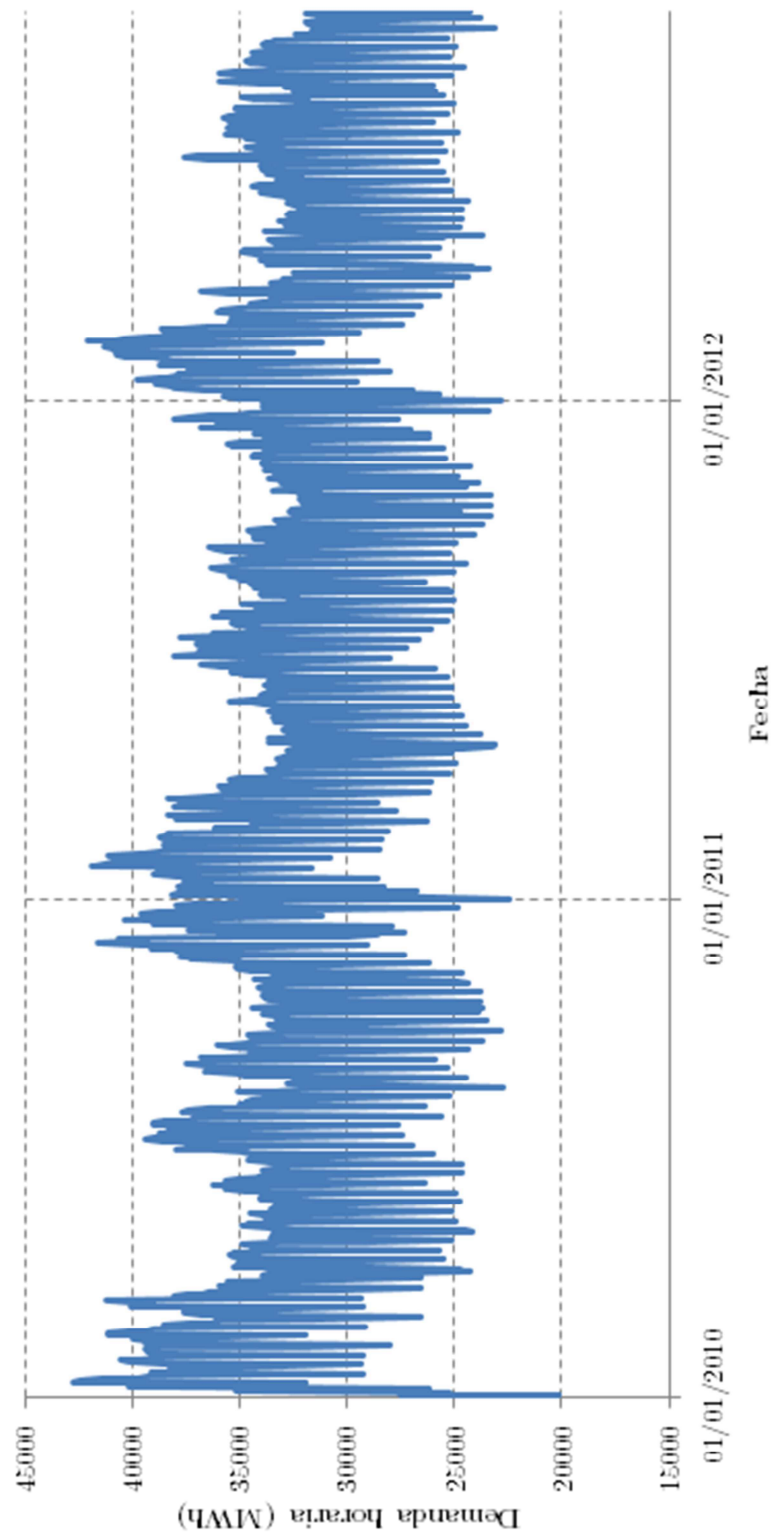


Figura 19. Demandas horarias reales 2010-2012

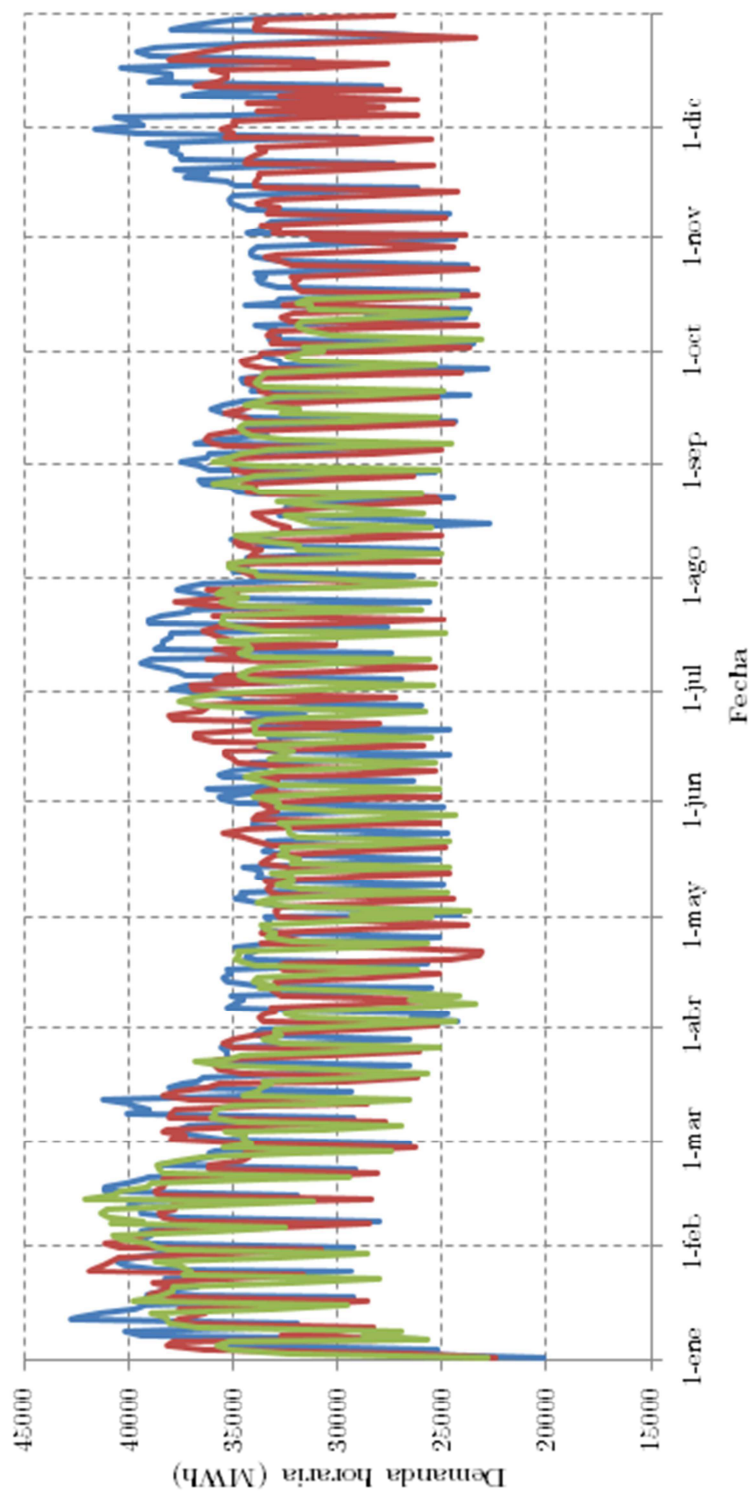


Figura 20. Superposición de demandas horarias reales 2010-2012

Como se puede observar de ambas gráficas, los datos de demanda horaria no presentan prácticamente ninguna tendencia y sí una clara estacionalidad. El modelo de predicción, por tanto, solo requerirá de un componente estacional, tal que:

$$D_{t,t+\tau}^* = a_t^* = \alpha D_t + (1 - \alpha)a_{t-1}^*$$

Siendo  $D_{t,t+\tau}^*$  la previsión de la demanda para el periodo  $t + \tau$ , hecha en el momento  $t$ ,  $a_t^*$  la estimación del componente estacionario al final del periodo  $t$ ,  $D_t$  la demanda real en el periodo  $t$  y  $\alpha$  la constante de alisado, comprendido entre 0 y 1.

No existe ninguna metodología que proporcione de forma rápida un valor adecuado para la constante de alisado, por lo que se realizó, tomando como referencia los datos del año 2009, la predicción para los años 2010 y 2011, con un  $\alpha$  distinto y calculamos la suma de todos los errores absoluto medio (MAD) y cuadrático medio (MSE) cometidos con respecto a los valores reales.

Los resultados obtenidos son los siguientes:

Alfa	MAD	MSE
0,1	3.067	17.682.256
0,2	2.964	16.358.843
0,3	2.875	15.366.813
0,4	2.799	14.653.344
0,5	2.737	14.173.311
0,6	2.692	13.889.289
0,7	2.663	13.771.549
0,8	2.648	13.798.065
0,9	2.657	13.954.504

Tabla 21. Errores de predicción.

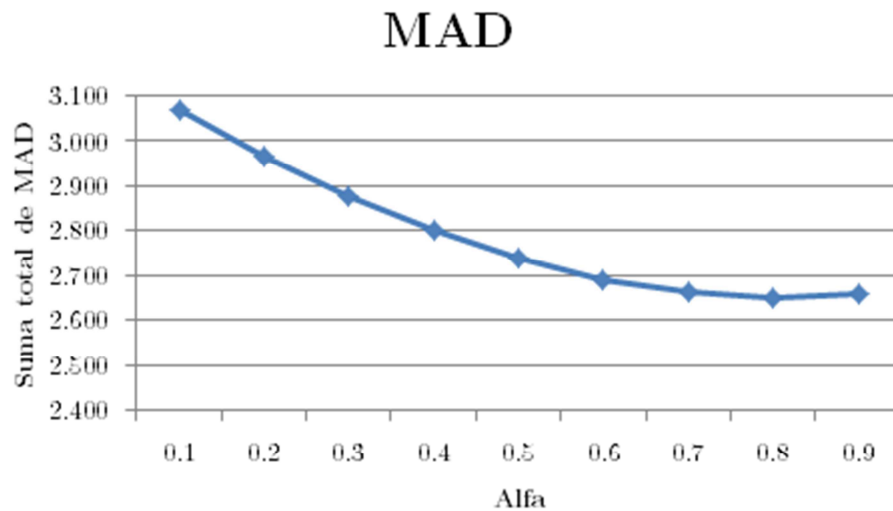


Figura 21. Gráfica de errores MAD.

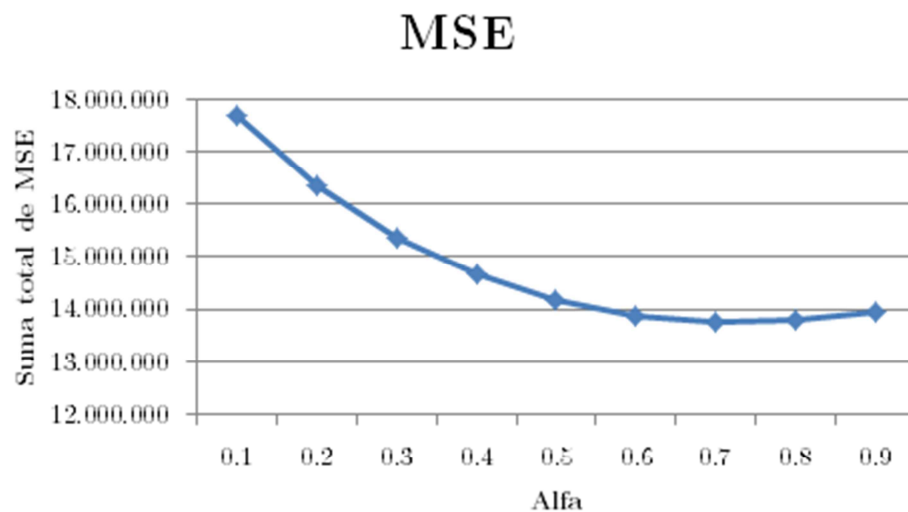


Figura 22. Gráfica de errores MSE

Como puede observarse en ambas gráficas, la suma de los errores absoluto medio y cuadrático medio acumulados se minimiza tomando un alfa de 0,75.

El modelo ha sido programado en PHP e incluido como librería al ‘banckend’ debido a la simplicidad de los cálculos, evitando tener que lanzar un ejecutable en Matlab, que reduciría el tiempo de ejecución de forma considerable.

#### 4.3.4 Generación de predicciones horarias

La predicción de precios horarios de energía eléctrica se realiza con la rutina disponible en el fichero PriceForecastUC3M.p.

La primitiva de la función es la siguiente:

```
[Ppred, Perc] = PriceForecastUC3M(Prices,nFor,log(Demands),Holidays,  
Parameter1, Parameter2, Array);
```

Los argumentos que se detallan son:

- **Prices:** Es una variable vector-columna con 8760 precios horarios en céntimos de €/KWh, es decir, los 24 precios horarios del año anterior, 365 días.
- **nFor:** Es una variable escalar que indica el horizonte temporal de predicción, es decir, el número de precios horarios a predecir. En este caso, son 144 datos de predicción que el proyecto necesita, esto son 6 días de precios horarios.
- **Demands:** Es una variable vector-columna con 8904 datos de demanda de energía eléctrica, en MWh, de los 365 días anteriores y la predicción de REE para los próximos 6 días.
- **Holidays:** Es una variable vector-columna con 8904 valores que indican si los datos de precio y demanda para cada hora corresponden con un día festivo o no.
- **Parameter 1 y 2:** Son parámetros de ajuste.
- **Array:** Es una variable vector fila que indica los límites de los intervalos de confianza, usualmente [2.5 5 10 50 80 90 97.5].

La función devuelve dos vectores:

- **Ppred:** Es un vector con las predicciones de los precios horarios obtenidos por la función, en céntimos de €/KWh.
- **Perc:** Es una matriz de 7 columnas, ya que el array [2.5 5 10 50 80 90 97.5] es 7x1, con los valores de los límites de los intervalos de confianza, en céntimos de €/KWh.

Los datos de precios y demandas que sirven como input a la rutina Matlab se obtienen de los datos acumulados obtenidos de OMIE y REE. Los datos del vector “Holidays” se calculan a partir de las fechas de festivos publicados en el BOE.

#### 4.3.5 Generación de predicciones mensuales y pico

La predicción de precios mensuales y de pico de energía eléctrica se realiza con la rutina disponible en el fichero MonthlyPriceForecastUC3M.p.

La primitiva de la función es la siguiente:

```
[PpredAR, PercAR, Escenarios] = MonthlyPriceForecastUC3M (Prices,  
nFor, [], Demands, Parameter1, Parameter2, Probs);
```

Los argumentos que se detallan son:

- **Prices:** Es una variable vector-columna de dimensión 84x1 con los 12 precios mensuales, es decir, los precios mensuales de los 7 años anteriores.
- **nFor:** Es una variable escalar que indica el horizonte temporal de predicción, es decir, los meses a predecir. En este caso, son 12 datos de predicción que el proyecto necesita, esto es un año de predicción.
- **Demands:** Es una variable vector-columna de dimensión 96x1 con los datos de demanda de energía eléctrica de los 84 meses anteriores y la predicción de REE para los próximos 12 meses.
- **Parameter 1 y 2:** Son parámetros de ajuste.
- **Probs:** Es una variable vector fila de dimensión kx1 que contiene los intervalos de confianza a estudio. Usualmente, k=5 y [5 10 25 50 75 90 95].

La función devuelve tres vectores:

- **PpredAR:** Es un vector con los datos de predicción de precios horarios obtenidos por la función.
- **PercAR:** Es una matriz de k columnas con la predicción de precios horarios a distintos intervalos de confianza.
- **Escenarios:** Datos sin interés para este proyecto.

Los datos de precios y demandas se los que se alimenta la rutina Matlab se obtienen de los datos acumulados obtenidos de OMIE y REE.

#### 4.3.6 Gestión de datos de predicciones

Los resultados de las rutinas Matlab se almacenan de forma persistente en la base de datos ‘priceforecasting’ en el servidor MySQL.

Paralelamente, estos mismos resultados se almacenan en el servidor de bases de datos NoSQL MongoDB. En el próximo capítulo se detalla su diseño.

#### 4.3.7 Cálculo de errores acumulados

Para el cálculo de los errores, se utiliza la fórmula de errores absolutos:

```
$this->_pred_[$i][$j] - $this->_RealPrice_[$j];
```

Donde la variable `$this->_pred_` contiene las predicciones y la variable `$this->_RealPrice_` los datos reales.

#### 4.3.8 Backup

Un “backup” es una copia de seguridad de datos informáticos para su posterior restauración en tres casos posibles:

- Recuperación de una catástrofe informática.
- Recuperación de archivos eliminados accidentalmente.
- Recuperación de archivos corrompidos.

En este proyecto se ha implementado un sistema de backup que realiza una copia de seguridad cada día a una hora determinada, de todos los datos en bruto del servidor MySQL. Este fichero, de entorno a los 18MB y aumentado a medida que el sistema almacena más datos, se almacena en el disco duro del servidor. Los ficheros “backup” están en formato SQL.

Debido a la falta de un servidor alternativo disponible, independiente del servidor empleado para la ejecución de este proyecto para el almacenamiento de los ficheros de datos “backup”, se ha diseñado un procedimiento de carga de estos ficheros a servicios de almacenamiento en la nube. El procedimiento se detalla a continuación.



En primer lugar se ejecuta la herramienta `mysqldump` con los parámetros:

```
mysqldump.exe --opt  
-h hostname nombre_base_de_datos  
-u nombre_de_usuario  
-p contraseña  
-r nombre_de_fichero_backup
```

El fichero resultante tendrá un nombre de la forma `backup_complete_AÑOMESDIA.sql`.

En segundo lugar, se realiza una carga del fichero a DropBox empleando la librería PHP puesta a disposición de los desarrolladores. Debido al reducido espacio de almacenamiento disponible en la cuenta, 2 GB, procedemos a contabilizar el número de ficheros “backup” en la cuenta para eliminar aquellos más antiguos, de tal forma que el número total de “backups” en DropBox no supere los 10 para no quedar sin espacio suficiente.

Finalmente, se realiza una carga del mismo fichero a Google Drive con la librería PHP desarrollada por Google para la misma. Por desgracia, al no estar la librería completamente desarrollada, no es posible de momento eliminar con sencillez ficheros ni actualizarlos, por lo que subiremos los ficheros “backup” sin tomar ninguna consideración.

## Capítulo 5

# Análisis, diseño e implementación del Frontend

### 5.1 Análisis

En la fase de análisis de la aplicación web, se ha empleado un diagrama de caso de uso (CU). Estos diagramas sirven para especificar la funcionalidad y comportamiento del sistema.

#### *5.1.1 Diagrama de caso de uso*

Un diagrama de caso de uso es una técnica con el objetivo de realizar una captura de requisitos de cualquier sistema informático. Cada caso muestra un escenario y en él que se indica cómo interactúan entre sí los usuarios y el sistema.

En este proyecto, constan tres tipos distintos de usuarios:

- **Usuario Esporádico:** Son aquellos usuarios no registrados en la aplicación web. Solo tienen acceso a las gráficas de predicción.
- **Usuario Normal:** Son aquellos usuarios registrados en la aplicación web. Este tipo de usuarios tienen acceso a las gráficas, los valores de predicción e incluso pueden descargarlos en formato Excel.
- **Administrador:** Son aquellos con permisos necesarios para llevar a cabo funciones administrativas y acceso con privilegios a perfiles de usuario.

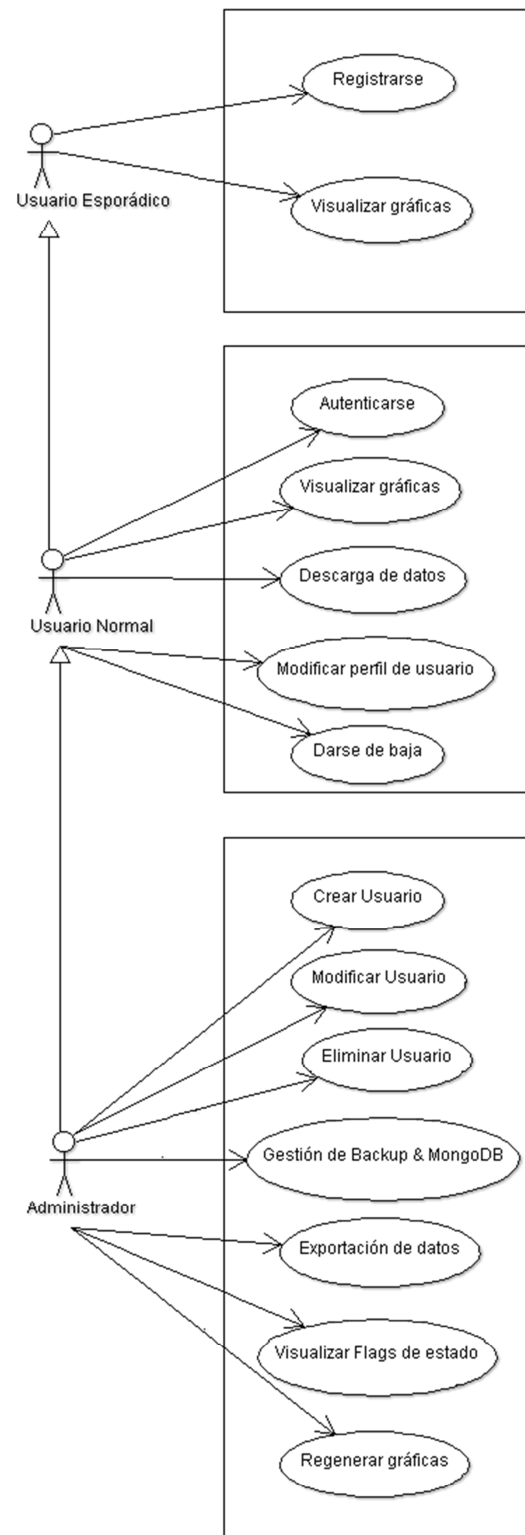


Figura 23. Diagrama de casos de uso

CU-01	
<b>Nombre</b>	Registrarse
<b>Actores</b>	Usuario Esporádico
<b>Precondiciones</b>	Ninguna
<b>Postcondiciones</b>	Usuario registrado
<b>Descripción</b>	
El sistema permite que un usuario no registrado pueda registrarse en la aplicación web.	
<b>Escenario</b>	
<p>Pulsar el botón 'Crear un cuenta'.</p> <p>Rellenar un formulario.</p> <p>Datos solicitados:</p> <ul style="list-style-type: none"> <li>▪ Nombre.</li> <li>▪ Apellidos.</li> <li>▪ Empresa u organización.</li> <li>▪ Nombre de usuario.</li> <li>▪ Contraseña.</li> <li>▪ Email.</li> </ul>	

Tabla 22. CU-01

CU-02	
<b>Nombre</b>	Visualizar gráficas
<b>Actores</b>	Usuario Esporádico, Usuario Registrado, Administrador
<b>Precondiciones</b>	Ninguna
<b>Postcondiciones</b>	Visualizar gráfica solicitada
<b>Descripción</b>	
El sistema permite que cualquier usuario pueda solicitar la gráfica que desea visualizar.	
<b>Escenario</b>	
<p>Acceder a la sección deseada a través del menú.</p> <p>Seleccionar una fecha.</p>	

Tabla 23. CU-02

CU-03	
<b>Nombre</b>	Autenticarse
<b>Actores</b>	Usuario Registrado, Administrador
<b>Precondiciones</b>	Usuario no ha iniciado sesión
<b>Postcondiciones</b>	Sesión de usuario iniciada
<b>Descripción</b>	
El sistema permite que usuarios registrados y administradores puedan iniciar sesión en la aplicación web.	
<b>Escenario</b>	
Pulsar 'Iniciar sesión' en el menú. Introducir nombre de usuario y contraseña. Resolver captcha. Pulsar el botón 'Iniciar Sesión'.	

Tabla 24. CU-03

CU-04	
<b>Nombre</b>	Descarga de datos
<b>Actores</b>	Usuario Registrado, Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Descarga de un fichero
<b>Descripción</b>	
El sistema permite que usuarios registrados y administradores puedan descargar los resultados de las predicciones y otros datos.	
<b>Escenario</b>	
Acceder a la sección deseada a través del menú. Escoger el formato de fichero deseado. Pulsa el botón correspondiente.	

Tabla 25. CU-04

CU-05	
<b>Nombre</b>	Modificar perfil de usuario
<b>Actores</b>	Usuario Registrado, Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Datos personales modificados
<b>Descripción</b>	
El sistema permite que usuarios registrados y administradores puedan modificar sus datos en su perfil de usuario.	
<b>Escenario</b>	
Acceder a la sección 'Perfil de Usuario' del menú. Modificar los datos personales.	

Tabla 26. CU-05

CU-06	
<b>Nombre</b>	Darse de baja
<b>Actores</b>	Usuario Registrado, Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Usuario eliminado
<b>Descripción</b>	
El sistema permite que usuarios registrados y administradores puedan darse de baja.	
<b>Escenario</b>	
Acceder a la sección 'Perfil de Usuario' del menú. Seleccionar opción de darse de baja. Confirmar selección. Sesión finalizada.	

Tabla 27. CU-06

CU-07	
Nombre	Crear usuario
Actores	Administrador
Precondiciones	Sesión de usuario iniciada
Postcondiciones	Usuario nuevo creado
<b>Descripción</b>	
El sistema permite que los administradores puedan registrar nuevos usuarios.	
<b>Escenario</b>	
Acceder a la sesión de administración. Rellenar un formulario. Datos solicitados: <ul style="list-style-type: none"><li>▪ Nombre.</li><li>▪ Apellidos.</li><li>▪ Empresa u organización.</li><li>▪ Nombre de usuario.</li><li>▪ Contraseña.</li><li>▪ Email.</li></ul>	

Tabla 28. CU-07

CU-08	
Nombre	Modificar usuario
Actores	Administrador
Precondiciones	Sesión de usuario iniciada
Postcondiciones	Datos y privilegios de usuario modificados
<b>Descripción</b>	
El sistema permite que los administradores puedan modificar los datos de un usuario, habilitar o deshabilitar acceso a la API y modificar el tipo de usuario.	
<b>Escenario</b>	
Acceder a la sesión de administración. Modificar datos personales Habilitar o deshabilitar acceso a API Modificar tipo de usuario.	

Tabla 29. CU-08

CU-09	
<b>Nombre</b>	Eliminar usuario
<b>Actores</b>	Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Usuario eliminado
<b>Descripción</b>	
El sistema permite que los administradores puedan eliminar usuarios.	
<b>Escenario</b>	
Acceder a la sesión de administración.	
Seleccionar usuario.	
Seleccionar opción de eliminar usuario.	

Tabla 30. CU-09

CU-10	
<b>Nombre</b>	Gestión de backup y MongoDB
<b>Actores</b>	Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Gestión de backups
<b>Descripción</b>	
El sistema permite que los administradores puedan generar un backup del servidor MySQL y MongoDB en un momento específico o importar una copia de seguridad en caso necesario.	
<b>Escenario</b>	
Acceder a la sesión de administración.	
Seleccionar la opción 'Regenerar Gráficos y MongoDB Collections'.	

Tabla 31. CU-10



CU-11	
<b>Nombre</b>	Exportación de datos
<b>Actores</b>	Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Descargar datos
<b>Descripción</b>	
El sistema permite que los administradores puedan descargar en formato Excel los datos en cualquiera de las tablas de la base de datos en MySQL.	
<b>Escenario</b>	
Acceder a la sesión de administración. Pulsar en la opción 'Exportar datos'. Seleccionar la tabla deseada.	

Tabla 32. CU-11

CU-12	
<b>Nombre</b>	Visualizar Flags de estado
<b>Actores</b>	Administrador
<b>Precondiciones</b>	Sesión de usuario iniciada
<b>Postcondiciones</b>	Visualizar listado de Flags
<b>Descripción</b>	
El sistema permite que los administradores puedan visualizar el estado de los flags usados internamente por la aplicación web para comprobar su estado.	
<b>Escenario</b>	
Acceder a la sesión de administración. Pulsar en la opción 'Visualizar status'.	

Tabla 33. CU-12

CU-13	
Nombre	Regenerar gráficas
Actores	Administrador
Precondiciones	Sesión de usuario iniciada
Postcondiciones	Gráficas regeneradas
Descripción	
El sistema permite que los administradores puedan eliminar las gráficas acumuladas en el servidor y volver a regenerarlas.	
Escenario	
Acceder a la sesión de administración. Pulsar en la opción 'Regenerar Gráficos y MongoDB Collections'.	

Tabla 34. CU-12

## 5.2 Diseño

### 5.2.1 Modelo de la base de datos

El gestor de bases de datos utilizado para la aplicación web del proyecto es MongoDB. Es un gestor NoSQL por lo que a priori no presenta vulnerabilidades a ataques SQL presentes en bases de datos relacionales.

Para la interfaz de la aplicación web con MongoDB, habiéndose instalado correctamente el driver para PHP proporcionado por la empresa desarrolladores 10gen Inc, se ha desarrollado un “wrapper” para facilitar el desarrollo del código del “frontend”.

Este “wrapper”, o interfaz, persigue un objetivo más, y es el de la seguridad. En el capítulo 3 se describen posibles ataques NoSQL. El “wrapper” desarrollado permite realizar una comprobación del tipo de dato en la consulta antes de realizarla en el gestor, por lo que eliminamos uno de los principales vectores de ataque.

Es importante resaltar que MongoDB almacena los datos en documentos, a diferencia de las bases de datos relacionales.

La base de datos en la que se almacenan los resultados de las previsiones se denomina “priceforecast”. Esta base de datos está a su vez compuesta por 6 colecciones:

- **omie-cassation-prices**

Almacena los precios horarios de casación publicados por OMIE.

Los documentos presentan una estructura de la forma:

```
{
  "_id" : ObjectId("4ff6051c4aebf6f818000000"),
  "date" : "2012-07-06",
  "hour" : "1",
  "portugal" : "5.613",
  "spain" : "5.613"
}
```

Las claves 'portugal' y 'spain' corresponden a los precios de casación, en céntimos de €/KWh, de Portugal y España, respectivamente.

- **short-term-historical-hour-forecast**

Almacena las predicciones de precios horarios realizadas por el sistema.

Los documentos presentan una estructura de la forma:

```
{
  "_id" : ObjectId("500886a54aebf6341c012a1f"),
  "date" : "2012-05-20",
  "hour" : "24",
  "forecast" : "4.5198",
  "perc1" : "2.4500",
  "perc2" : "2.7231",
  "perc3" : "3.0936",
  "perc4" : "4.5249",
  "perc5" : "5.5997",
  "perc6" : "6.2201",
  "perc7" : "7.2547",
  "datestamp" : "2012-05-13"
}
```

La clave 'forecast' corresponde a la predicción de precios horarios en céntimos de €/KWh. Las claves 'perc1', 'perc2', 'perc3', 'perc4', 'perc5', 'perc6' y 'perc7' corresponden a los límites de los intervalos de confianza 5, 10, 25, 50, 75, 90 y 95, respectivamente. La clave 'datestamp' indica la fecha en la que se realizó la predicción.

- **ree-esios-forecasted-month-demand**

Almacena las previsiones de demanda publicados por REE.

Los documentos presentan una estructura de la forma:

```
{
  "_id" : ObjectId("4ff60a3d4aebf6f818001da0"),
  "year" : "2012",
}
```

```
"month" : "6",  
"demand" : "20.428",  
"datestamp" : "2012-05"  
}
```

La clave 'demand' corresponde a la demanda prevista en España en TWh, mientras que la clave 'datestamp' indica el mes y año en el que se realizó la predicción.

- **long-term-historical-hour-forecast**

Almacena las predicciones de precios mensuales realizadas por el sistema.

Los documentos presentan una estructura de la forma:

```
{  
  "_id" : ObjectId("4ff6f4354aebf6ac27000000"),  
  "datestamp" : "2010-01-01",  
  "forecast" : "3.65858",  
  "year" : "2010",  
  "month" : "1",  
  "perc1" : "2.67722",  
  "perc2" : "2.93944",  
  "perc3" : "3.35146",  
  "perc4" : "3.72514",  
  "perc5" : "4.01537",  
  "perc6" : "4.33431",  
  "perc7" : "4.39511"  
}
```

La clave 'forecast' corresponde a la predicción de precios mensuales en céntimos de €/KWh. Las claves 'perc1', 'perc2', 'perc3', 'perc4', 'perc5', 'perc6' y 'perc7' corresponden a los límites de los intervalos de confianza 5, 10, 25, 50, 75, 90 y 95, respectivamente. La clave 'datestamp' indica la fecha en la que se realizó la predicción.

- **peak-historical-forecast**

Almacena las predicciones de precios pico realizadas por el sistema.

Los documentos presentan una estructura de la forma:

```
{  
  "_id" : ObjectId("4ff6f6354aebf6582400000c"),  
  "datestamp" : "2010-02-01",  
  "forecast" : "3.2299",  
  "year" : "2010",  
  "month" : "2",  
  "perc1" : "3.2299",  
  "perc2" : "3.2299",  
}
```

```
"perc3" : "3.2299",  
"perc4" : "3.2299",  
"perc5" : "3.2299",  
"perc6" : "3.2299",  
"perc7" : "3.2299"  
}
```

La clave 'forecast' corresponde a la predicción de precios mensuales en céntimos de €/KWh. Las claves 'perc1', 'perc2', 'perc3', 'perc4', 'perc5', 'perc6' y 'perc7' corresponden a los límites de los intervalos de confianza 5, 10, 25, 50, 75, 90 y 95, respectivamente. La clave 'datestamp' indica la fecha en la que se realizó la predicción.

#### ▪ user-control

Almacena los datos de los usuarios registrados en la aplicación web.

Los documentos presentan una estructura de la forma:

```
{  
  "_id" : ObjectId("500409654aebf64815080000"),  
  "user-api" : "1",  
  "user-company" : "UC3M",  
  "user-email" : "jexsk84@gmail.com",  
  "user-id" : "58c0ee27e3d74c8c3cc1dcc79a6690c3",  
  "user-level" : "admin",  
  "user-login" : "Pricefor",  
  "user-name" : "Pricefor",  
  "user-pass" : "e10adc3949ba8abbe56e057f20f883d2",  
  "user-surname" : " ",  
  "user-token" : "7c41b857ba484ec3e5540177ce8105f2"  
}
```

Entre las claves disponibles en los documentos de esta colección, cabe señalar la clave 'user-api', que indica si el usuario tiene acceso a la api; 'user-id', que es un código alfanumérico único que identifica al usuario; 'user-level', que indica si el usuario es normal 'normal' o administrador 'admin' con mayores privilegios; y la clave 'user-token', que es usada en el mecanismo de identificación de usuarios en la api.

## 5.3 Implementación

Procedemos a documentar los módulos en los que está construida la aplicación web.

### 5.3.1 *Diseño de la interfaz web*

El interfaz web de este proyecto se ha construido mediante lenguaje PHP que genera HTML en su ejecución, junto con hojas de estilo CSS. Para ello, se ha utilizado un “framework” CSS desarrollado por Twitter denominado Twitter Bootstrap.

Un “framework” CSS es una librería preparada para mejorar la estandarización de páginas web de forma fácil y cómoda. En ellas se encuentran soluciones estandarizadas para conseguir un mismo diseño en cualquier navegador web.

Twitter Bootstrap, en concreto, permite la creación de diseños web combinando CSS con JavaScript. Ofrece una serie de plantillas CSS y ficheros JavaScript para integrar el framework de forma sencilla y potente en cualquier proyecto web.

Destacan funcionalidades como:

- Crear interfaces tanto para escritorio como tablets y móviles con distintas resoluciones.
- Se integra completa con la librería JQuery.
- Diseño basado en estándares CSS3 y HTML5
- Funciona en todos los navegadores, incluido Internet Explorer.

Entre las ventajas de uso de un “framework” CSS destacan:

- Permite agilizar el desarrollo de una aplicación web. Ahorra tiempo y esfuerzo.
- Dispone de un soporte multinavegador.
- Suelen ser proyectos públicos con una gran comunidad involucrada en su desarrollo.

Los razones fundamentales del uso de Twitter Bootstrap en este proyecto son la necesidad de llevar un control del código estándar y la necesidad de evitar introducir modificaciones en el código CSS para mejorar la visualización de la interfaz en navegadores antiguos.

### 5.3.2 Funcionalidad general

La aplicación web ha sido diseñada para ser la plataforma de interacción del usuario con el sistema. Cada una de las páginas web que la conforman está compuesta de un menú [A] en la parte superior, que da acceso a cuatro grupos:

- **Predicciones Horarias.** Compuesta por dos páginas: “Predicción para los próximos 7 días” y “Predicción y Percentiles” con las predicciones de precios horarios.
- **Predicciones Mensuales.** Compuesta por dos páginas: “Predicción próximos meses” y “Predicción y Percentiles” con las predicciones de precios mensuales.
- **Predicciones Pico.** Compuesta por dos páginas: “Predicción próximos meses” y “Predicción y Percentiles” con las predicciones de precios pico.
- **Errores.** Compuesta por tres páginas: “Errores Precios Horarios”, “Errores Precios Mensuales”, “Errores Precios Pico”, que muestran los errores acumulados para cada predicción.



Figura 24. Página "dashboard" de la interfaz web

Cada página incluida en los tres primeros grupos consta a su vez de otras tres secciones:

- **Gráfico.** Esta sección está formada por un calendario desplegable [A] desde el que poder seleccionar la fecha deseada y un gráfico con las predicciones [B].

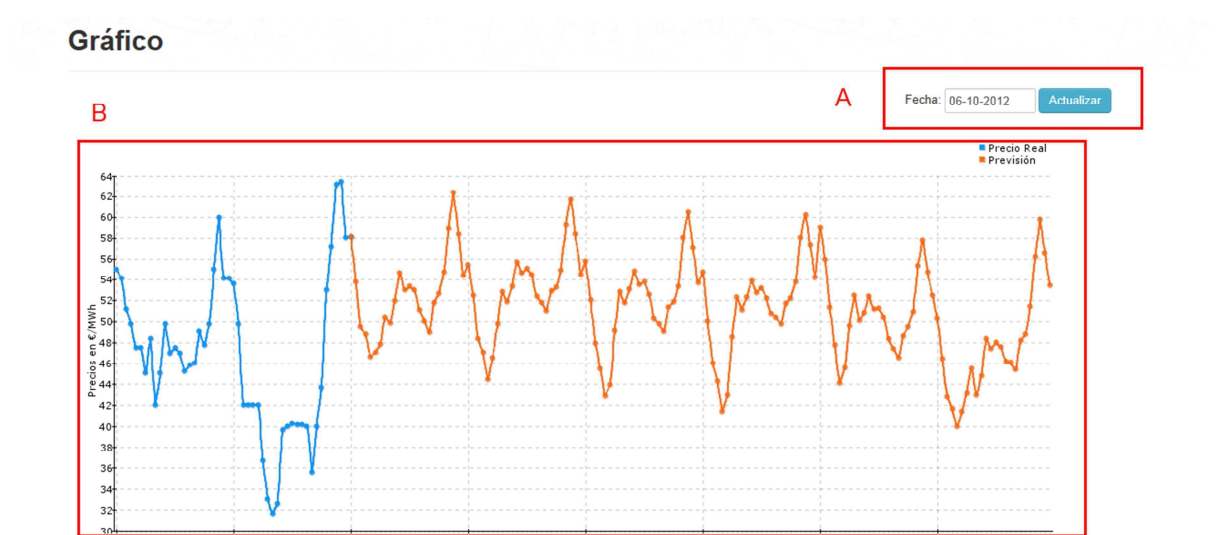


Figura 25. Sección "Gráfico" de la interfaz web

- **Resultados.** Esta sección está formada por, al menos, un mensaje informativo [A] que proporciona al usuario información relevante y una tabla [B] con las predicciones.

**Resultados**

Tabla/s de datos mostrados en la gráfica.

**A**

Datos en €/MWh

**B**

Fecha	Hora	Precio
08-10-2012	1	58.087
08-10-2012	2	53.724
08-10-2012	3	49.458
08-10-2012	4	48.744
08-10-2012	5	46.617
08-10-2012	6	47.082
08-10-2012	7	47.823
08-10-2012	8	50.357
08-10-2012	9	49.857
08-10-2012	10	51.892
08-10-2012	11	54.65

Figura 26. Sección "Resultados" de la interfaz web



- **Descargas.** Esta sección está formada por un panel [A] que contiene 3 botones. El botón [B] permite la descarga de los datos en formato Excel tradicional, el botón [C] permite la descarga en formato Excel 2007/2010, y el botón [D] permite la descarga en forma de fichero de texto plano con los valores separados por comas.

## Descargas

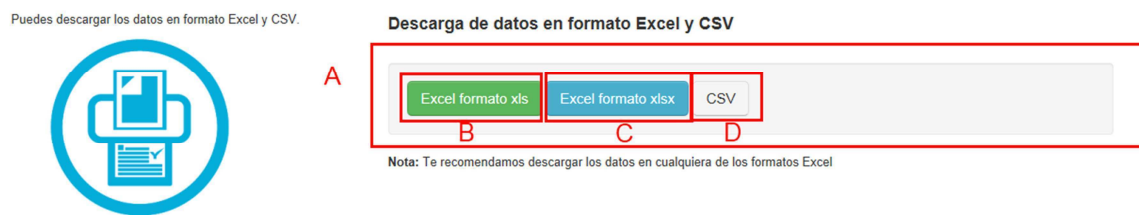


Figura 27. Sección "Descargas" de la interfaz web

Las páginas incluidas en el cuarto grupo, "Errores" únicamente contienen un gráfico [A].

## Gráfico

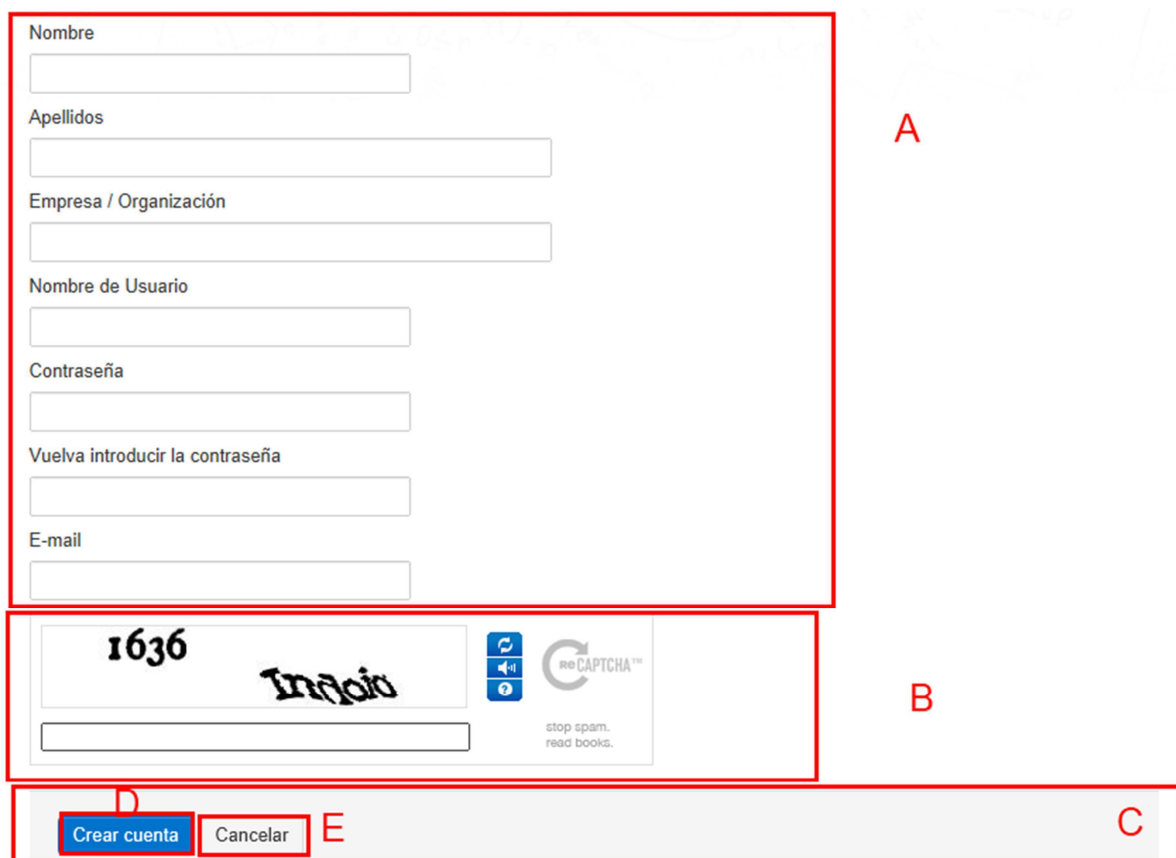


Figura 28. Sección "Gráfico" en las páginas del grupo "Errores"

Las secciones descritas no están disponibles para todos los usuarios. Tal y como se describe en el diagrama de casos de uso, existen tres tipos distintos de usuario: esporádico, normal y administrador. Los usuarios esporádicos solo tienen acceso a la sección "Gráfico" de cada página. En cambio, los usuarios normales y administradores tienen acceso a las tres secciones.

### 5.3.3 Registro de usuarios

Cualquier usuario puede registrarse en la web. Para ello, la página “dashboard” incluye un botón con el nombre “Crear una cuenta”. Al pulsarlo, el usuario es trasladado a un formulario [A] en el que debe introducir su nombre, apellidos, empresa u organización a la que pertenecen, un nombre de usuario, contraseña y un email. Todos los campos son obligatorios. En la misma página aparece un “captcha” [B] para evitar que se creen usuarios de forma automática por un bot, programa informático diseñado para crear cuentas de usuario en servicios web y publicar mensajes de spam, y un panel [C] con los botones “Crear cuenta” [D] y “Cancelar” [E].



The image shows a web registration form with several labeled components:

- A**: A large red box enclosing the main registration form fields: "Nombre", "Apellidos", "Empresa / Organización", "Nombre de Usuario", "Contraseña", "Vuelva introducir la contraseña", and "E-mail".
- B**: A red box enclosing the CAPTCHA section, which includes a grid of numbers and letters (1636, Inoio), a reCAPTCHA logo, and a text input field.
- C**: A red box enclosing the bottom navigation bar, which contains two buttons: "Crear cuenta" and "Cancelar".
- D**: A red box specifically highlighting the "Crear cuenta" button.
- E**: A red box specifically highlighting the "Cancelar" button.

Figura 29. Página "register" de la interfaz web

### 5.3.4 Identificación de usuarios

En la barra de menú de la parte superior de la interfaz web, aparece un enlace para permitir a los usuarios autenticarse. Al pulsarlo, el usuario es trasladado a la página “login” compuesta por dos campos [A] en los que usuario debe introducir su nombre de usuario y

contraseña, y un panel [B] compuesto a su vez de tres botones: “Iniciar Sesión” [C], “Cancelar” [D] y “Recuperar Contraseña” [E].

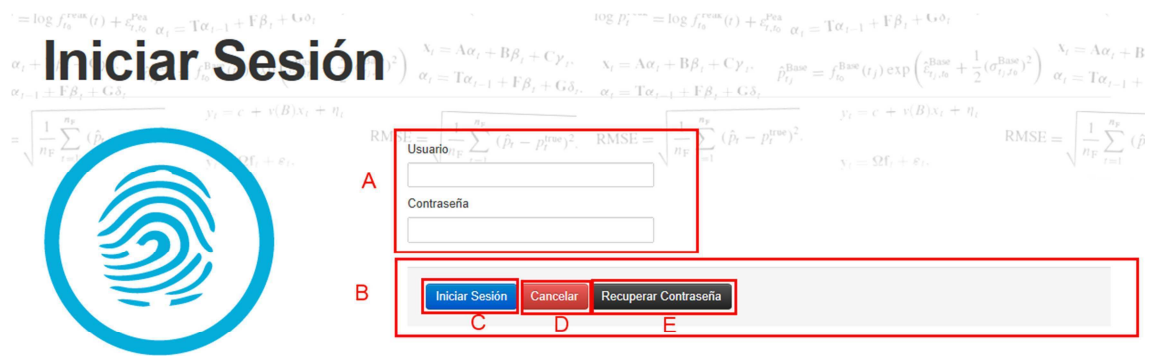


Figura 30. Página "login" de la interfaz web

En el caso de que un usuario olvide la contraseña, el botón F da acceso a un nuevo formulario compuesto por un único campo [A] en el que el usuario debe introducir su cuenta de correo electrónico con el que se registró, un “captcha” [B] y un panel [C] compuesto por los botones “Recuperar Contraseña” [D] y “Cancelar” [E]. Una vez pulsado el botón D, y solo si existe un usuario registrado con esa cuenta de correo electrónico, se le manda al mismo una nueva contraseña generada aleatoriamente.



Figura 31. Página "recover" de la interfaz web

Una vez el usuario se ha autenticado, en el menú superior aparece un nuevo submenú [A] con su nombre de usuario que da acceso a la página “Perfil de Usuario”.



Figura 32. Menú de usuario

La página "Perfil de Usuario" permite al usuario modificar sus datos personales, a excepción de su nombre de usuario. Incluye asimismo, la posibilidad de regenerar el token que le proporciona acceso a la API [A], muestra si tiene acceso a la misma [B], el tipo de usuario en el que está clasificado [C] y la permite la posibilidad de darse de baja [D].

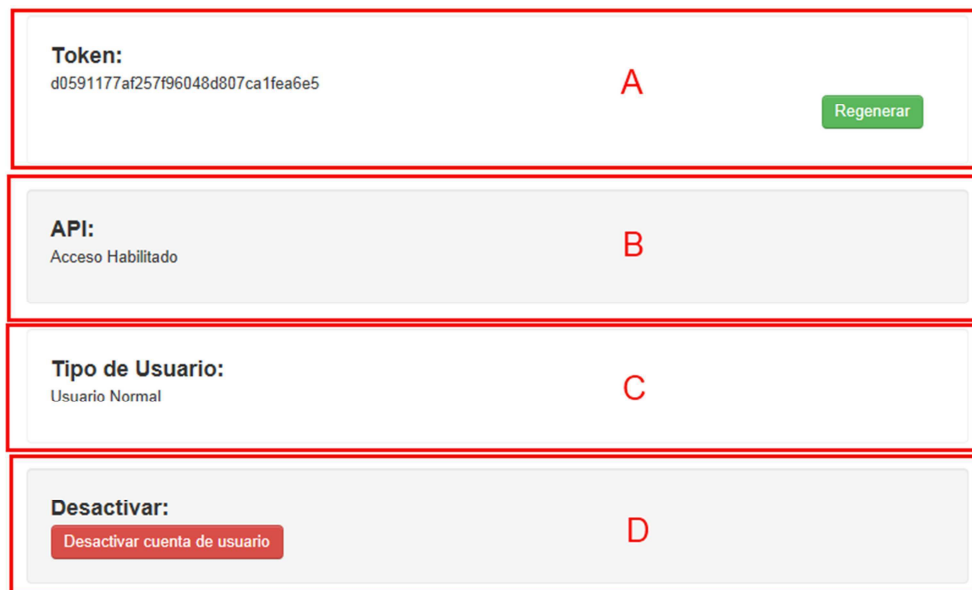


Figura 33. Página "profile" de la interfaz web

### 5.3.5 Administración de la aplicación web

Los usuarios administradores tienen acceso exclusivo a las siguientes páginas:

- **Predicciones Horarias > Precios Horarios.** Se muestran los precios de casación publicados por OMIE.
- **Predicciones Horarias > Demandas Horarias.** Se muestra la previsión de demanda horaria publicada por REE.
- **Predicciones Mensuales > Demandas Mensuales Previstas.** Se muestran las demandas mensuales que predice REE.

En el menú de usuario, también se incluye un nuevo apartado: “Administración”.



Figura 34. Administración interfaz web

El panel de administración está compuesto de 6 secciones:

- **“Visualizar Status”.** Permite comprobar el estado en el que se encuentra el sistema.
- **“Gestión de Usuarios”.** Permite gestionar los usuarios registrados.
- **“Gestión y Descarga de Backups”.** Permite gestionar los “backups” acumulados.



- **“Gestión de festivos”**. Permite añadir y eliminar fechas de festivos.
- **“Importar un Backup”**. Permite restaurar el sistema a partir de un “backup”.
- **“Regenerar Gráficos y MongoDB Collections”**. Permite volver a regenerar las gráficas e importar datos al gestor de base de datos MongoDB.
- **“Etiquetas de Texto”**. Permite modificar el texto que se visualiza en la interfaz web.
- **“Exportar Datos”**. Permite descargar los datos almacenados en la base de datos.

## Capítulo 6

# API HTTP/RESTful

### 6.1 Descripción general

Una API, “Application Programming Interface”, es un conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Representa la capacidad de comunicación entre componentes de software.

REST, “Representational State Transfer”, es un término que se originó en el año 2000 por Roy Fielding, uno de los principales autores de la especificación del protocolo HTTP. Describe cualquier interfaz web simple que utiliza XML y HTTP sin las abstracciones usadas en servicios web SOAP.

Este tipo de interfaces están diseñados bajo los siguientes criterios:

- **Un protocolo cliente-servidor sin estado:** Las peticiones se realizan usando el protocolo HTTP, que contiene toda la información necesaria para ejecutarla, por lo que ambos agentes no necesitan recordar el estado de las comunicaciones entre mensajes.
- **Uso de métodos HTTP:** Se requiere en la mayoría de los casos de los métodos GET, POST, PUT y DELETE presentes en el protocolo HTTP/1.1.
- **Uso de XML:** Para el intercambio eficiente de sistemas heterogéneos, se utilizan formato extensibles como XML y HTTP.

En el caso de la API implementada en este proyecto, ésta se basa en el protocolo HTTP/1.0 ya que únicamente requiere el método GET. Está diseñada para ser en modo lectura y la interacción con el usuario es mínima. En ningún momento, la API permite realizar modificaciones en los datos o eliminarlos.

## 6.2 Estructura

La API está compuesta principalmente de un “handler”, que es quien gestiona las peticiones de los clientes, que se corresponde con la URL:

`http://pricefor.uc3m.es/api/`

El “handler” maneja únicamente peticiones realizadas por el método GET, por lo que los parámetros que definen la consulta deben estar incluidos en el método, tal y como define su RFC.

El modelo implementado en la API es cliente-servidor. El cliente abre una conexión al servidor y realiza la petición. El servidor, procesa la respuesta, la transmite al cliente y finaliza la conexión.

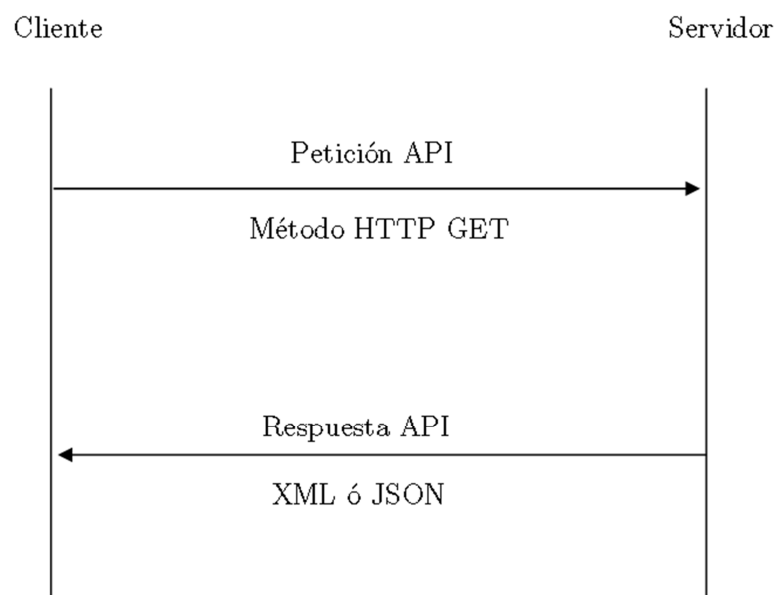


Figura 35. Modelo API Cliente-Servidor

La API acepta ocho parámetros distintos, que combinados, forman una consulta:

- **token**

El token es una cadena alfanumérica aleatoria. Generada al registrarse un usuario en la aplicación web, cada usuario dispone de un token completamente único. Esta cadena permite a la API identificar al usuario que realiza la consulta. La aplicación



web permite modificar este token en caso necesario. Es un parámetro obligatorio en cada petición.

- **format**

El parámetro 'format' indica el formato de los datos que desea el cliente obtener.

Solo admite 2 posibles valores:

- xml: Descrito en el capítulo 3.
- json: Es un formato de intercambio de información alternativo a xml.

Es un parámetro obligatorio en cada petición.

- **data\_type**

El parámetro 'data\_type' indica el tipo de consulta.

Solo admite los siguientes valores:

- omie\_prices: Para consultas de precios horarios de casación.
- hour\_forecast: Para consultas de predicciones de precios horarios.
- hour\_confidence: Para consultas de predicciones de precios horarios e intervalos de confianza.
- month\_demands: Para consultas de predicciones de demanda mensual.
- month\_forecast: Para consultas de predicciones de precios mensuales
- month\_confidence: Para consultas de predicciones de precios mensuales e intervalos de confianza.
- peak\_forecast: Para consultas de predicciones de precios pico.
- peak\_confidence: Para consultas de predicciones de precios pico e intervalos de confianza.

Es un parámetro obligatorio en cada petición.

- **start\_date**

El parámetro 'start\_date' indica la fecha de predicción o inicio de rango de datos a consultar. Emplea el formato de fecha ISO 8601 AAAA-MM-DD para datos y/o predicciones horarias.

Es un parámetro obligatorio para las peticiones: 'omie\_prices', 'hour\_forecast' y 'hour\_confidence'.

- **finish\_date**

El parámetro 'finish\_date' indica la fecha de fin de rango de datos a consultar. Emplea el formato de fecha ISO 8601 AAAA-MM-DD para datos y/o predicciones horarias.

Es un parámetro opcional para las peticiones: 'omie\_prices', 'hour\_forecast' y 'hour\_confidence'.

- **execution\_date**

El parámetro 'execution\_date' indica la fecha de ejecución de la predicción. Es únicamente válido y opcional para las peticiones 'hour\_forecast' y 'hour\_confidence'.

- **year**

El parámetro 'year' indica el año de predicción.

Es únicamente válido para las peticiones 'month\_forecast', 'month\_confidence', 'peak\_forecast' y 'peak\_confidence', y es un parámetro obligatorio en las peticiones 'month\_forecast', 'month\_confidence', 'peak\_forecast' y 'peak\_confidence'.

- **month**

El parámetro 'month' indica el mes de predicción.

Es únicamente válido para las peticiones 'month\_forecast', 'month\_confidence', 'peak\_forecast' y 'peak\_confidence', y es un parámetro obligatorio en las peticiones 'month\_forecast', 'month\_confidence', 'peak\_forecast' y 'peak\_confidence'.

### 6.3 Ejemplos de uso

Obtener valores de las predicciones horarias para el día 6 de julio de 2012 en formato xml.

```
http://pricefor.uc3m.es/api/  
?token=7c41b857ba484ec3e5540177ce8105f2  
&data_type=hour_forecast  
&start_date=2012-07-06  
&format=xml
```

Obtener valores de las predicciones horarias para el día 6 de julio de 2012 en formato json.

```
http://pricefor.uc3m.es/api/  
?token=7c41b857ba484ec3e5540177ce8105f2  
&data_type=hour_forecast  
&start_date=2012-07-06  
&format=json
```

Obtener valores de las predicciones mensuales para julio de 2012 en formato json.

```
http://pricefor.uc3m.es/api/  
?token=dala8ba41d6a36910  
&data_type=month_forecast  
&year=2011  
&month=09  
&format=json
```

## 6.4 Aplicación “PriceForecasting Client”

Como ejemplo de uso de la API del proyecto, se ha desarrollado un cliente denominado “PriceForecasting Client”.

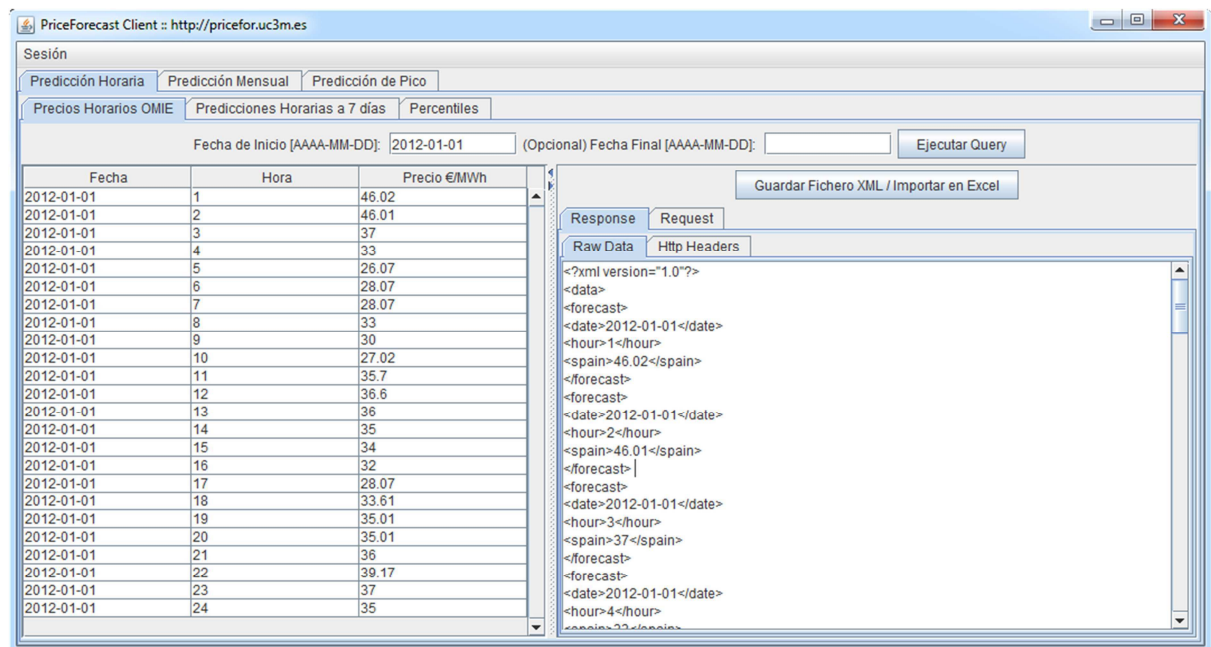


Figura 36. PriceForecast Client

Incorpora una interfaz de usuario que incluye de forma fácil y sencilla todas las opciones y parámetros que permite la API.

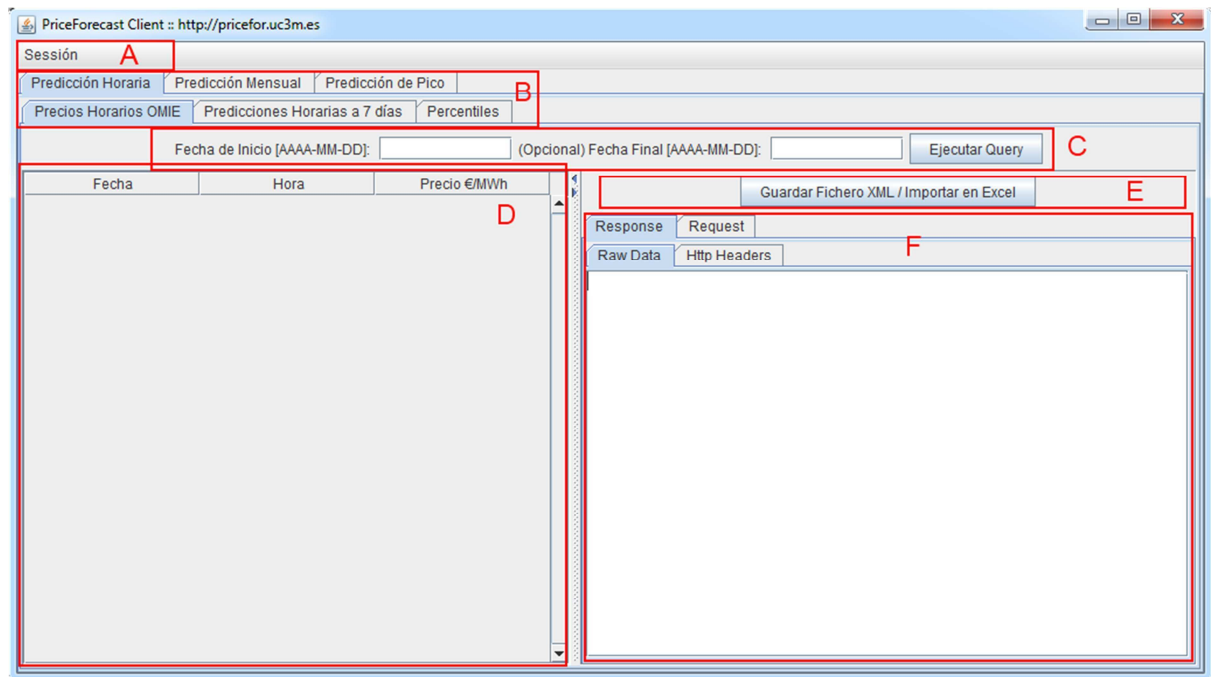


Figura 37. Interfaz cliente API

Está compuesta de las siguientes secciones:

- **A:** Menú para iniciar y finalizar la sesión.
- **B:** Conjunto de pestañas para acceder a cada tipo de datos y a las predicciones disponibles.
- **C:** Permite introducir parámetros para construir la consulta.
- **D:** Muestra en una tabla los resultados de la consulta.
- **E:** Permite guardar los datos en formato XML.
- **F:** Visualiza las cabeceras HTTP y el resultado de la consulta en bruto.

En primer lugar, el usuario debe introducir su token accediendo al menú 'Sesión'. La API dispone de un mecanismo de comprobación de credenciales, especificando en el "handler" los parámetros 'check' y 'token'.

```
http://pricefor.uc3m.es/api/  
?check  
&token=7c41b857ba484ec3e5540177ce8105f2
```

El API retorna el nombre de usuario correspondiente y en caso de error, el resultado es '-1'. La aplicación almacenará en memoria este token y lo incluirá automática en cada consulta que el cliente realice.

Una vez autenticado, la sección C incluye para cada pestaña, parámetros que el usuario debe introducir junto con algunos parámetros opcionales señalados como tal. Al pulsar el botón 'Ejecutar Query', la aplicación realiza la consulta al servidor y espera a los resultados.

La sección D muestra los resultados en formato tabla.

Fecha	Hora	Precio €/MWh
2012-01-01	1	46.02
2012-01-01	2	46.01
2012-01-01	3	37
2012-01-01	4	33
2012-01-01	5	26.07
2012-01-01	6	28.07
2012-01-01	7	28.07
2012-01-01	8	33
2012-01-01	9	30
2012-01-01	10	27.02
2012-01-01	11	35.7
2012-01-01	12	36.6
2012-01-01	13	36
2012-01-01	14	35
2012-01-01	15	34
2012-01-01	16	32
2012-01-01	17	28.07
2012-01-01	18	33.61
2012-01-01	19	35.01
2012-01-01	20	35.01
2012-01-01	21	36
2012-01-01	22	39.17
2012-01-01	23	37
2012-01-01	24	35

Figura 38. Sección D de "PriceForecast Client"

Los datos internos de cada consulta se muestran en la sección F. Esta sección está compuesta de dos pestañas principales: 'Request' y 'Response'.

La pestaña 'Request' muestra la construcción del método GET del protocolo HTTP con el que se realiza la consulta.

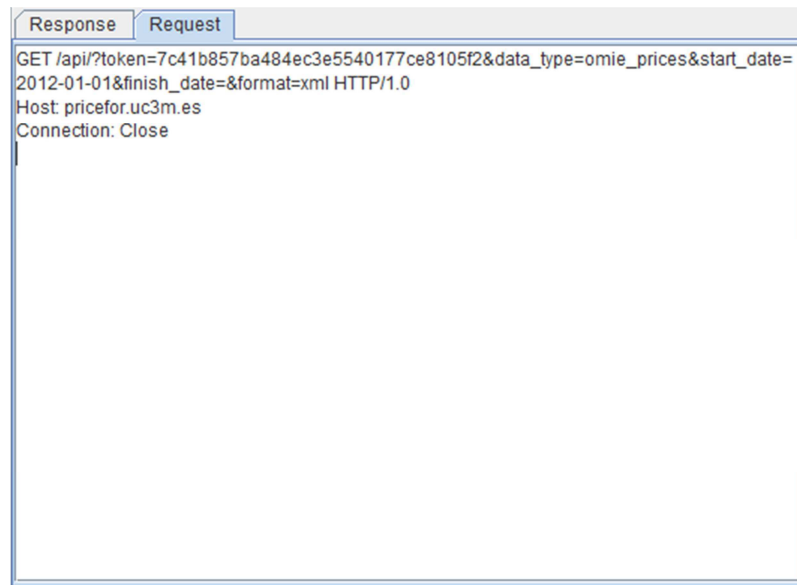


Figura 39. Pestaña 'Request' de "PriceForecast Client"

La pestaña 'Response' a su vez incluye dos pestañas: 'Http Headers' y 'Raw Data'. La pestaña 'Http Headers' muestra la cabecera HTTP de la respuesta del servidor. Permite comprobar si la respuesta se corresponde con una respuesta correcta '200 OK', o con errores '404 Not Found', '500 Internal Server Error' o '501 Not Implemented'.

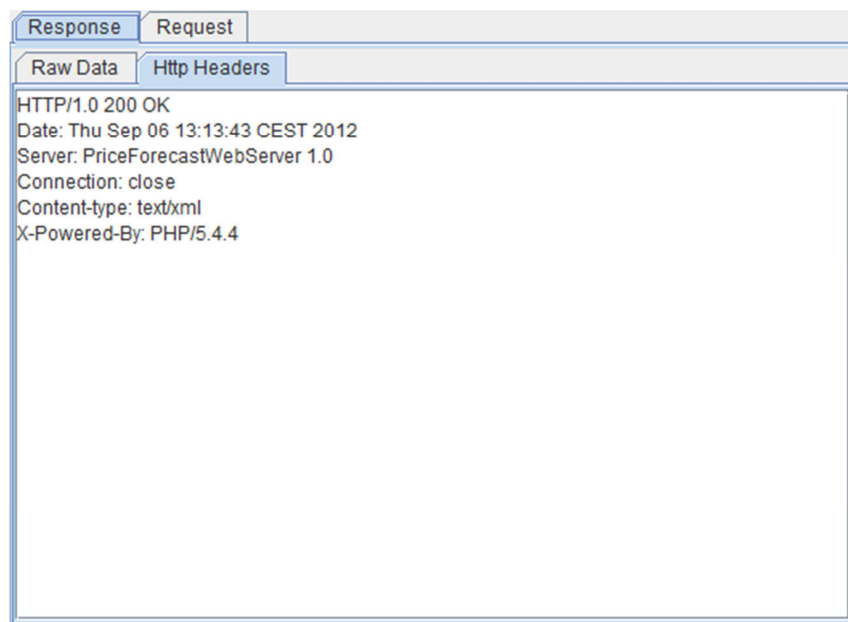


Figura 40. Pestaña 'Http Headers' de "PriceForecast Client"

La pestaña 'Raw Data' muestra los datos en bruto resultado de la consulta.

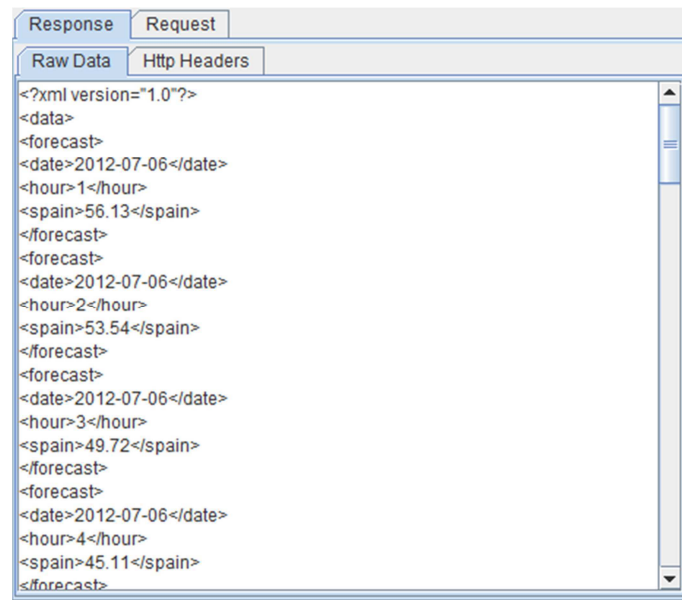


Figura 41. Pestaña 'Raw Data' de "PriceForecast Client"

## Capítulo 7

### Evaluación

#### 7.1 MySQL – PHP vs. MySQL ODBC Driver

Matlab incorpora un mecanismo para medir el tiempo de ejecución de una rutina, con los comandos `tic` y `toc`. Este apartado evaluará el tiempo de ejecución necesario para realizar una prueba con los comandos ‘INSERT’ para introducir en la base de datos 500 valores y posteriormente ejecutar el comando ‘SELECT’ para obtener de nuevo esos mismos valores; esto es, una prueba de escritura y lectura en la base de datos. Se realizan 5 pruebas.

Los resultados obtenidos con MySQL – PHP son los siguientes:

Test	Tiempo de ejecución (s)
1	1,226212
2	1,442109
3	1,061518
4	1,436036
5	1,159618

Tabla 35. Resultados tests PHP-MySQL



Los resultados obtenidos con MySQL ODBC Driver son los siguientes:

Test	Tiempo de ejecución (s)
1	17,127350
2	17,073431
3	17,121893
4	17,116593
5	16,679536

Tabla 36. Resultados tests MySQL ODBC Driver

A partir de los resultados podemos comprobar que la pasarela MySQL-PHP desarrollada para Matlab es significativamente más rápida que el driver que proporciona Oracle. La media de ejecución usando MySQL-PHP es de 1,265 segundos mientras que la media de ejecución usando el driver ODBC es de 17,02 segundos, es decir, este último es alrededor de 13 veces más lento.

## 7.2 Web Server Stress Tests

Los ‘Web Server Stress Tests’ permiten predecir el comportamiento de un servicio web en condiciones aproximadas a las de producción, es decir, con el sistema en pleno funcionamiento tras su desarrollo. Para ello hacemos uso de la herramienta Apache Benchmark (AB).

### 7.2.1 Caso I

Tomamos como primer caso una carga de 1000 usuarios y 5 usuarios accediendo a la interfaz web de forma concurrente. Se realizaron 5 pruebas consecutivas y los resultados son los siguientes:

### 7.2.1.1 Apache

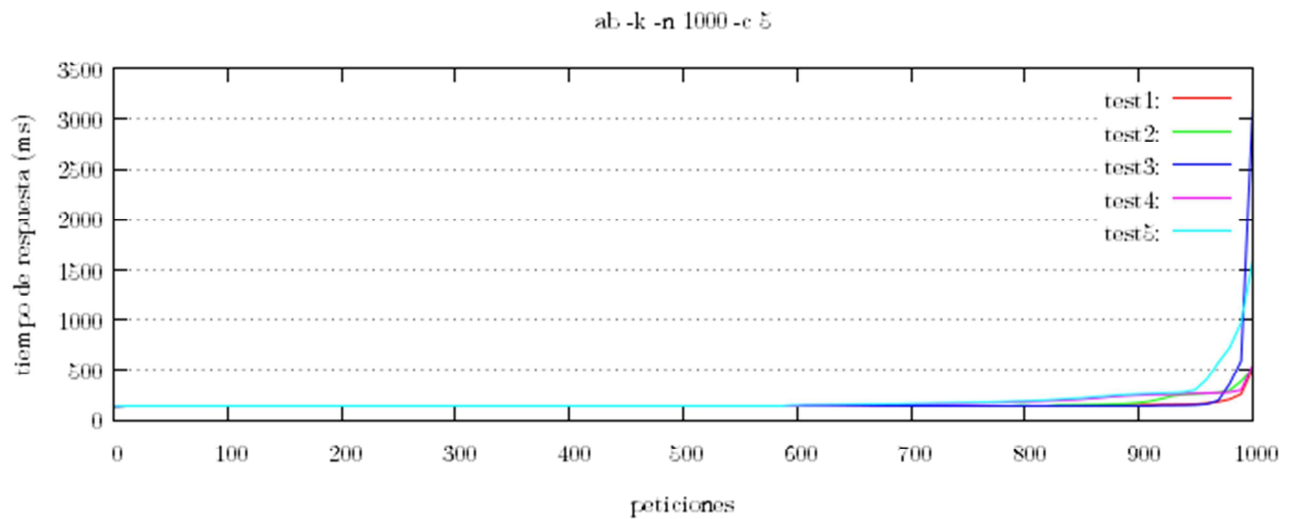


Figura 42. Gráfica resultados AB. Apache. Caso I

Porcentaje	Tiempo de respuesta (ms)
50%	141
66%	143
75%	146
80%	152
90%	193
95%	259
98%	291
99%	515
100%	3.095

Tabla 37. Tabla de resultados AB. Apache. Caso I

### 7.2.1.2 PriceForecast Web Server

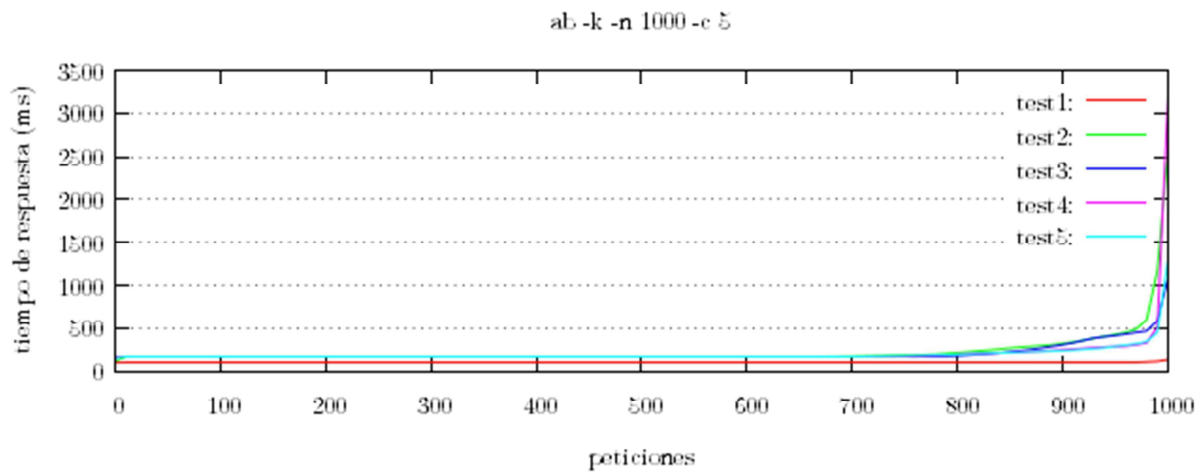


Figura 43. Gráfica resultados AB. PriceForecast Web Server. Caso I

Porcentaje	Tiempo de respuesta (ms)
50%	100
66%	101
75%	101
80%	101
90%	102
95%	103
98%	106
99%	114
100%	133

Tabla 38. Tabla de resultados AB. PriceForecast Web Server. Caso I

### 7.2.1.3 Análisis

De los resultados observamos que el tiempo de respuesta, usando Apache, es menor o igual a 193 milisegundos en el 90% de las peticiones, y menor o igual a 515 milisegundos en el 99% de las peticiones. Son tiempos de respuesta muy rápidos.

En cambio, usando PriceForecast Web Server, el tiempo de respuesta es menor o igual a 102 milisegundos en el 90% de las peticiones y menor o igual a 114 milisegundos en el 99%. Son tiempos de respuesta extraordinariamente rápidos.

### 7.2.2 Caso II

Tomamos como segundo caso una carga de 5000 usuarios y 20 usuarios accediendo a la interfaz web de forma concurrente. Se realizaron 5 pruebas consecutivas y los resultados son los siguientes:

#### 7.2.2.1 Apache

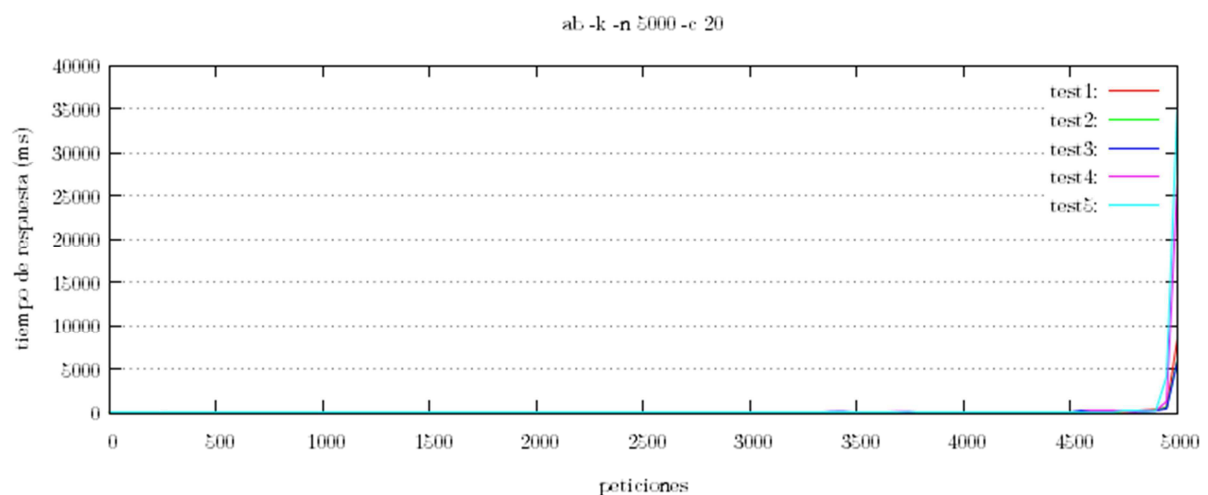


Figura 44. Gráfica resultados AB. Apache. Caso II

Porcentaje	Tiempo de respuesta (ms)
50%	151
66%	159
75%	179
80%	188
90%	194
95%	202
98%	315
99%	566
100%	35.287

Tabla 39. Tabla de resultados AB. Apache. Caso II

#### 7.2.2.2 *PriceForecast Web Server*

PriceForecast Web Server no pudo completar la prueba y por tanto no hay resultados disponibles.

#### 7.2.2.3 *Análisis*

De los resultados observamos que el tiempo de respuesta, usando Apache, es menor o igual a 194 milisegundos en el 90% de las peticiones, y menor o igual a 566 milisegundos en el 99% de las peticiones. Son tiempos de respuesta rápidos.

En cambio, el servidor PriceForecast no es capaz de soportar esta carga de trabajo.

### 7.2.3 Caso III

Tomamos como tercer caso una carga de 5000 usuarios y 100 usuarios accediendo a la interfaz web de forma concurrente. Se realizaron 5 pruebas consecutivas y los resultados son los siguientes:

#### 7.2.3.1 Apache

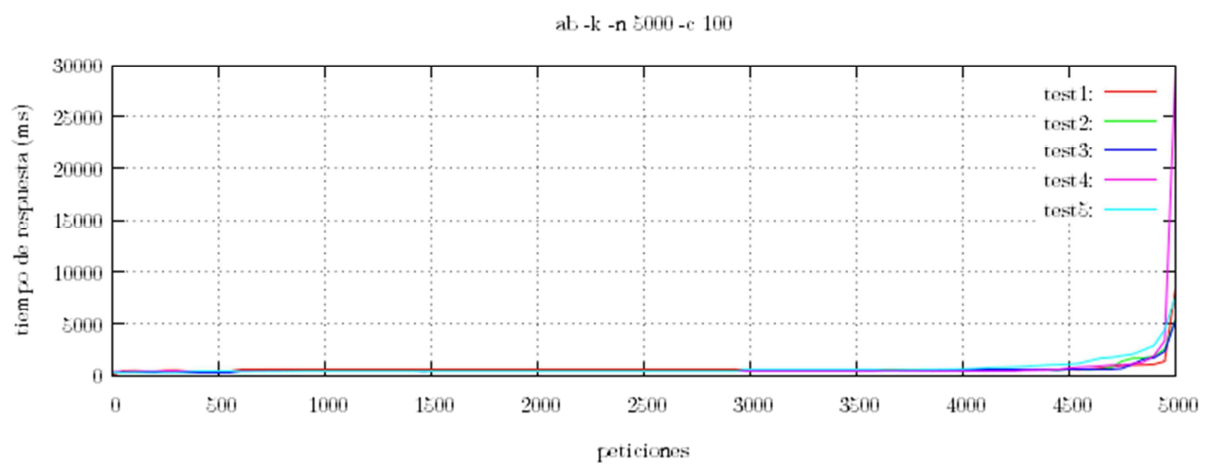


Figura 45. Gráfica resultados AB. Apache. Caso III

Porcentaje	Tiempo de respuesta (ms)
50%	448
66%	501
75%	521
80%	526
90%	602
95%	1.063
98%	1.850
99%	2.948
100%	29.124

Tabla 40. Tabla de resultados AB. Apache. Caso III



### **7.2.3.2 PriceForecast Web Server**

PriceForecast Web Server no pudo completar la prueba y por tanto no hay resultados disponibles.

### **7.2.3.3 Análisis**

De los resultados observamos que, en este tercer caso, el tiempo de respuesta, usando Apache, es menor o igual a 602 milisegundos en el 90% de las peticiones, y menor o igual a 1063 milisegundos en el 98% de las peticiones. Son tiempos de respuesta aceptables.

El servidor PriceForecast es incapaz de atender a cualquier petición con un elevado número de usuarios accediendo de forma concurrente.

### 7.3 Web-Based Exploitation

El análisis de vulnerabilidades “web-based” en el que se centra este apartado, se basa en la existencia de vulnerabilidades en la interfaz web usando el servidor Apache. El sistema hace uso de la versión 2.4.3. de Apache para Windows y es actualmente la última versión estable disponible.

Para el análisis de vulnerabilidades se ha procedido a ejecutar la herramienta Nikto usando BackTrack Linux, una distribución de Linux diseñada para la auditoria de seguridad informática. El único input que requiere Nikto es la ip del servidor a estudio.

El resultado es el siguiente:

```
>> perl nikto.pl -h 163.117.132.169

- Nikto v2.1.5
-----
+ Target IP:          163.117.132.169
+ Target Hostname:    163.117.132.169
+ Target Port:        80
+ Start Time:         2012-10-03 16:53:02 (GMT-4)
-----
+ Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
+ Retrieved x-powered-by header: PHP/5.4.7

+  DEBUG  HTTP  verb  may  show  server  debugging  information.  See
http://msdn.microsoft.com/en-us/library/e8z0lxdh%28VS.80%29.aspx for details.
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to
XST
+ OSVDB-3268: /config/: Directory indexing found.
+ /config/: Configuration information may be available remotely.
+ OSVDB-48: /doc/: The /doc/ directory is browsable. This may be /usr/doc.
+ OSVDB-682: /webalizer/: Webalizer may be installed. Versions lower than 2.01-
09 vulnerable to Cross Site Scripting (XSS).
+ OSVDB-12184: /index.php?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals
potentially sensitive information via certain HTTP requests that contain
specific QUERY strings.
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3092: /client/: This might be interesting...
+ OSVDB-3268: : Directory indexing found.
+ OSVDB-3268: /lib/: Directory indexing found.
+ OSVDB-3092: /lib/: This might be interesting...
+ OSVDB-3093: /admin/index.php: This might be interesting... has been seen in
web logs from an unknown scanner.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3268: /images/?pattern=/etc/*&sort=name: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ OSVDB-3092: /Admin/: This might be interesting...
+ /login.php: Admin login page/section found.
+ /phpmyadmin/: phpMyAdmin directory found
+ 6474 items checked: 1 error(s) and 21 item(s) reported on remote host
+ End Time:          2012-10-03 17:11:40 (GMT-4) (1118 seconds)
-----
+ 1 host(s) tested
```



De los resultados obtenidos se deduce la existencia de directorios accesibles en el servidor que podrían contener información sensible y de utilidad para un ataque. Concretamente, Nikto ha detectado que al acceder a carpetas como 'config', Apache genera un listado de ficheros de forma automática al no detectar la existencia de ningún fichero denominado 'index.html' o 'index.php'. Además, se han detectado directorios tales como 'webalizer' y 'phpmyadmin' en el que se ejecutan herramientas diversas para el desarrollo de la interfaz web que en fase de producción no deben ser accesibles por motivos de seguridad.

Se procede a subsanar los mencionados posibles vectores de ataque y a realizar una nueva prueba. El resultado es el siguiente:

```
>> perl nikto.pl -h 163.117.132.169

- Nikto v2.1.5
-----
+ Target IP:          163.117.132.169
+ Target Hostname:    pricefor.uc3m.es
+ Target Port:        80
+ Start Time:         2012-10-05 17:13:04
-----
+ Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
+ OSVDB-0: Retrieved X-Powered-By header: PHP/5.4.7
+ OSVDB-48: GET /doc/ : The /doc/ directory is browsable. This may be /usr/doc.
+ OSVDB-877: TRACE / : TRACE option appears to allow XSS or credential theft.
See      http://www.cgisecurity.com/whitehat-mirror/WhitePaper_screen.pdf      for
details
+ OSVDB-12184: GET /index.php?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000 : PHP
reveals potentially sensitive information via certain HTTP requests which
contain specific QUERY strings.
+ OSVDB-3092: GET /admin/ : This might be interesting...
+ OSVDB-3092: GET /client/ : This might be interesting...
+ OSVDB-3093: GET /admin/index.php : This might be interesting... has been seen
in web logs from an unknown scanner.
+ 3577 items checked: 8 item(s) reported on remote host
+ End Time:           2012-10-05 17:21:34 (510 seconds)
-----
+ 1 host(s) tested
```

Estos nuevos resultados permiten confirmar que la herramienta, tras realizar las modificaciones oportunas en la interfaz web, no detecta ninguna vulnerabilidad potencial.

## 7.4 Port & Vulnerability Scanning

Para realizar con éxito un ataque y posterior intrusión en un servidor web, el atacante debe conocer todos los servicios en ejecución que se comunican con la red a través de un puerto tcp o udp específico. Generalmente, la mayoría de servicios tienen asignado un puerto por defecto. Es el caso de los servidores web, que se comunican a través del puerto TCP 80 y TCP 443 si se usan conexiones cifradas; de los servidores de correo SMTP, a través del puerto TCP 25; y de servidores MySQL, a través del puerto TCP 3306.

### 7.4.1 Nmap

La herramienta más utilizada para identificar puertos abiertos es nmap. Para ello, nmap realiza una conexión a cada uno de los puertos para comprobar si están abiertos o cerrados, y en caso de que estén abiertos, procede a identificar el servicio.

#### 7.4.1.1 TCP Connect Scan

Se procede a ejecutar esta herramienta para realizar un escaneado de todos los puertos TCP, un total de 65535. El resultado es el siguiente:

```
>> nmap -sT -p- -PN 163.117.132.169

Starting Nmap 6.01 ( http://nmap.org ) at 2012-10-02 08:14 EDT
Nmap scan report for pricefor.uc3m.es (163.117.132.169)
Host is up (0.050s latency).
Not shown: 64508 closed ports, 1018 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
443/tcp    open  https
1041/tcp   open  danf-ak2
3306/tcp   open  mysql
3389/tcp   open  ms-wbt-server
10000/tcp  open  snet-sensor-mgmt
27017/tcp  open  unknown
28017/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 435.90 seconds
```

A la vista de los resultados se puede observar que nmap ha detectado la presencia de un servidor web ejecutándose en los puertos 80 y 443, un servidor MySQL en el puerto 3306, el 'MS Terminal Server RDP Client' en el puerto 3389 con el que se realiza el acceso en remoto al servidor, el 'Network Data Management Protocol' en el puerto 10000; así como otros no identificados.

Todos estos servicios pueden contener errores en su programación que permiten ser usados como vectores de ataque y obtener acceso al servidor. Por ello, deben estar permanentemente actualizados a fin de evitar posibles intrusiones.

#### 7.4.1.2 *TCP SYN Scan*

Se procede a realizar un escaneado de todos los puertos TCP usando el método ‘SYN Scan’. El resultado es el siguiente:

```
>> nmap -sS -p- -PN 163.117.132.169

Starting Nmap 6.01 ( http://nmap.org ) at 2012-10-02 08:26 EDT
Interesting ports on pc-132-169.uc3m.es (163.117.132.169):
Not shown: 64508 closed ports, 1018 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
443/tcp    open  https
3306/tcp   open  mysql
3389/tcp   open  ms-term-serv
10000/tcp  open  snet-sensor-mgmt
17500/tcp  open  unknown
27017/tcp  open  unknown
28017/tcp  open  unknown
```

Este método de escaneado de puertos ha localizado un único servicio adicional que etiqueta como desconocido en el puerto 17500.

#### 7.4.1.3 *UDP Scan*

A priori, el servidor no tiene en ejecución ningún servicio que requiera de la transferencia de paquetes usando el protocolo UDP. Realizando un escaneado de puertos mediante nmap y usando como método el ‘UDP Scan’, comprobamos que efectivamente es así.

```
>> nmap -sUV 163.117.132.169

Starting Nmap 6.01 ( http://nmap.org ) at 2012-10-02 08:35 EDT
Nmap scan report for pricefor.uc3m.es (163.117.132.169)
Host is up (0.053s latency).
All 1000 scanned ports on pricefor.uc3m.es (163.117.132.169) are closed (898)
or open|filtered (102)

Service detection performed. Please report any incorrect results at
http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 502.42 seconds
```

#### 7.4.1.4 Xmas Scan

Debido a que ningún de las versiones del S.O. Windows, tanto en versión doméstica como para servidores, sigue de forma fiel la RFC del estándar tcp/ip, ese método de escaneo de puertos no proporciona información alguna.

#### 7.4.2 Nessus

Otra de las herramientas más utilizadas para el escaneo de puertos es Nessus. A diferencia de nmap, Nessus incorpora toda una serie de 'plugins' que dotan a esta herramienta de la capacidad de analizar el servicio que está ejecutándose en cada puerto.

A través de su interfaz web, configuramos dos modos distintos de análisis:

- **Análisis básico.** Incluye un escaneo de todos los puertos tcp y udp.
- **Análisis avanzado.** Amplía el análisis básico incluyendo todos los plugins de los que dispone Nessus para identificar vulnerabilidades en el servidor.

Como situación de partida, hemos de considerar que el S.O. Windows Server 2003 que se ejecuta en el servidor no ha sido actualizado en los últimos meses. Esto permitirá comprobar que Nessus puede detectar posibles vías de ataque en aquellos sistemas con software no actualizado.

Los resultados del análisis básico son los siguientes:

10335 (5) - Nessus TCP scanner
<b>Synopsis</b>
It is possible to determine which TCP ports are open.
<b>Description</b>
This plugin is a classical TCP port scanner. It shall be reasonably quick even against a firewalled target. Once a TCP connection is open, it grabs any available banner for the service identification plugins. Note that TCP scanners are more intrusive than SYN (half open) scanners.
<b>Solution</b>
Protect your target with an IP filter.
<b>Risk Factor</b>
None
<b>Hosts</b>
163.117.132.169 (tcp/80)
Port 80/tcp was found to be open
163.117.132.169 (tcp/443)
Port 443/tcp was found to be open
163.117.132.169 (tcp/3306)
Port 3306/tcp was found to be open
163.117.132.169 (tcp/3389)
Port 3389/tcp was found to be open
163.117.132.169 (tcp/10000)
Port 10000/tcp was found to be open

Figura 46. Nessus. Análisis básico

Como se puede observar, el análisis básico usando Nessus es similar al obtenido usando el primer método con la herramienta nmap.

Los resultados del análisis avanzado son los siguientes:

163.117.132.169					
Summary					
Critical	High	Medium	Low	Info	Total
0	1	14	1	30	46
Details					
Severity	Plugin Id	Name			
High (9.3)	58435	MS12-020: Vulnerabilities in Remote Desktop Could Allow Remote Code Execution (2671387) (uncredentialed check)			
Medium (6.9)	62101	Apache 2.2 < 2.2.23 Multiple Vulnerabilities			
Medium (6.4)	51192	SSL Certificate Cannot Be Trusted			
Medium (6.4)	57582	SSL Self-Signed Certificate			
Medium (5.1)	18405	Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness			
Medium (5.0)	10677	Apache mod_status /server-status Information Disclosure			
Medium (5.0)	10678	Apache mod_info /server-info Information Disclosure			
Medium (5.0)	20007	SSL Version 2 (v2) Protocol Detection			
Medium (5.0)	46803	PHP expose_php Information Disclosure			
Medium (5.0)	57791	Apache 2.2 < 2.2.22 Multiple Vulnerabilities			
Medium (4.3)	11213	HTTP TRACE / TRACK Methods Allowed			
Medium (4.3)	26928	SSL Weak Cipher Suites Supported			
Medium (4.3)	42873	SSL Medium Strength Cipher Suites Supported			
Medium (4.3)	57792	Apache HTTP Server httpOnly Cookie Information Disclosure			
Medium (4.3)	58751	SSL/TLS Protocol Initialization Vector Implementation Information Disclosure Vulnerability			
Low (2.6)	26194	Web Server Uses Plain Text Authentication Forms			
Info	10107	HTTP Server Type and Version			
Info	10114	ICMP Timestamp Request Remote Date Disclosure			
Info	10287	Traceroute Information			
Info	10335	Nessus TCP scanner			
Info	10662	Web mirroring			
Info	10863	SSL Certificate Information			
Info	10940	Windows Terminal Services Enabled			
Info	11032	Web Server Directory Enumeration			
Info	11219	Nessus SYN scanner			

Figura 47. Nessus. Análisis avanzado I

Info	11936	OS Identification
Info	12053	Host Fully Qualified Domain Name (FQDN) Resolution
Info	14788	IP Protocols Scan
Info	20109	Trend Micro OfficeScan Client Detection
Info	21643	SSL Cipher Suites Supported
Info	21740	Trend Micro OfficeScan Client Version
Info	22964	Service Detection
Info	24260	HyperText Transfer Protocol (HTTP) Information
Info	25220	TCP/IP Timestamps Supported
Info	39463	HTTP Server Cookies Set
Info	42057	Web Server Allows Password Auto-Completion
Info	43111	HTTP Methods Allowed (per directory)
Info	45410	SSL Certificate commonName Mismatch
Info	45590	Common Platform Enumeration (CPE)
Info	49704	External URLs
Info	49705	Web Server Harvested Email Addresses
Info	51891	SSL Session Resume Supported
Info	54615	Device Type
Info	56984	SSL / TLS Versions Supported
Info	57041	SSL Perfect Forward Secrecy Cipher Suites Supported
Info	57323	OpenSSL Version Detection

Figura 48. Nessus. Análisis avanzado I (continuación)

Como se puede observar del listado de resultados, Nessus ha detectado una vulnerabilidad que clasifica como 'HIGH' que permite mediante "Servicios de Escritorio Remoto" ejecutar código en el servidor sin necesidad de autenticación. En el boletín de seguridad para MS12-020, publicado por Microsoft en marzo de 2012, se la denomina como "Vulnerabilidades en Escritorio remoto podrían permitir la ejecución remota de código (2671387)" y se la califica como crítica.

Tras actualizar el S.O. realizamos el mismo análisis de nuevo. Los resultados son los siguientes:

163.117.132.169					
Summary					
Critical	High	Medium	Low	Info	Total
0	0	12	2	32	46
Details					
Severity	Plugin Id	Name			
Medium (6.4)	51192	SSL Certificate Cannot Be Trusted			
Medium (6.4)	57582	SSL Self-Signed Certificate			
Medium (5.1)	18405	Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness			
Medium (5.0)	10678	Apache mod_info /server-info Information Disclosure			
Medium (5.0)	45411	SSL Certificate with Wrong Hostname			
Medium (5.0)	46803	PHP expose_php Information Disclosure			
Medium (4.3)	11213	HTTP TRACE / TRACK Methods Allowed			
Medium (4.3)	26928	SSL Weak Cipher Suites Supported			
Medium (4.3)	31705	SSL Anonymous Cipher Suites Supported			
Medium (4.3)	42873	SSL Medium Strength Cipher Suites Supported			
Medium (4.3)	57690	Terminal Services Encryption Level is Medium or Low			
Medium (4.3)	58751	SSL/TLS Protocol Initialization Vector Implementation Information Disclosure Vulnerability			
Low (2.6)	26194	Web Server Uses Plain Text Authentication Forms			
Low (2.6)	30218	Terminal Services Encryption Level is not FIPS-140 Compliant			
Info	10107	HTTP Server Type and Version			
Info	10114	ICMP Timestamp Request Remote Date Disclosure			
Info	10287	Traceroute Information			
Info	10335	Nessus TCP scanner			
Info	10662	Web mirroring			
Info	10863	SSL Certificate Information			
Info	10940	Windows Terminal Services Enabled			
Info	11032	Web Server Directory Enumeration			
Info	11219	Nessus SYN scanner			
Info	11919	HMAP Web Server Fingerprinting			
Info	11936	OS Identification			

Figura 49. Nessus. Análisis avanzado II

Info	12053	Host Fully Qualified Domain Name (FQDN) Resolution
Info	14788	IP Protocols Scan
Info	17975	Service Detection (GET request)
Info	20109	Trend Micro OfficeScan Client Detection
Info	21643	SSL Cipher Suites Supported
Info	21740	Trend Micro OfficeScan Client Version
Info	22319	MSRPC Service Detection
Info	22964	Service Detection
Info	24260	HyperText Transfer Protocol (HTTP) Information
Info	25220	TCP/IP Timestamps Supported
Info	39463	HTTP Server Cookies Set
Info	42057	Web Server Allows Password Auto-Completion
Info	43111	HTTP Methods Allowed (per directory)
Info	45410	SSL Certificate commonName Mismatch
Info	45590	Common Platform Enumeration (CPE)
Info	46180	Additional DNS Hostnames
Info	49704	External URLs
Info	54615	Device Type
Info	56984	SSL / TLS Versions Supported
Info	57041	SSL Perfect Forward Secrecy Cipher Suites Supported
Info	57323	OpenSSL Version Detection

Figura 50. Nessus. Análisis avanzado II (continuación)

A la vista de los resultados, la vulnerabilidad crítica comentada anteriormente ha desaparecido. Con respecto a posibles vulnerabilidades que Nessus califica como moderados y de mayor riesgo destacan los certificados SSL instalados en el servidor Apache que no proceden de una entidad certificadora reconocida y la posibilidad de un ataque ‘Man-in-the-middle’ usando el “Servicio de Escritorio Remoto” ya que esta versión de Windows Server permite la posibilidad de acceder a él sin cifrar la conexión.



## 7.5 Server Exploitation

A partir de los resultados obtenidos con nmap, un atacante puede considerar como principales vectores potenciales:

- **Apache Server.** Puertos 80 y 443.
- **Servidor MySQL.** Puerto 3306.
- **MS Terminal Server RDP Client.** Puerto 3389.

Para intentar penetrar en el servidor, la herramienta por excelencia, utilizada en la inmensa mayoría de las auditorías de seguridad, es Metasploit. Sin embargo, como Nessus, tras haber actualizado el S.O. no ha detectado ninguna vulnerabilidad crítica, no disponemos de un vector de ataque con el que utilizar Metasploit.

Por ello recurrimos a la herramienta ‘Fast-Track Autopwn’. Lanzamos esta herramienta y el resultado es el siguiente:

```
>> python fast-track.py --i

*****
**                                                                 **
** Fast-Track - A new beginning...                               **
** Version: 4.0.2                                                **
** Written by: David Kennedy (ReL1K)                             **
** Lead Developer: Joey Furr (j0fer)                             **
** http://www.secmaniac.com                                       **
**                                                                 **
*****

Fast-Track Main Menu:

1. Fast-Track Updates
2. Autopwn Automation
3. Nmap Scripting Engine
4. Microsoft SQL Tools
5. Mass Client-Side Attack
6. Exploits
7. Binary to Hex Payload Converter
8. Payload Generator
9. Fast-Track Tutorials
10. Fast-Track Changelog
11. Fast-Track Credits
12. Exit Fast-Track

Enter the number: 2

*****
**                                                                 **
** Fast-Track - A new beginning...                               **
** Version: 4.0.2                                                **
** Written by: David Kennedy (ReL1K)                             **
** Lead Developer: Joey Furr (j0fer)                             **
** http://www.secmaniac.com                                       **
**                                                                 **
*****
```



Metasploit Autopwn Automation:

`http://www.metasploit.com`

This tool specifically piggy backs some commands from the Metasploit Framework and does not modify the Metasploit Framework in any way. This is simply to automate some tasks from the autopwn feature already developed by the Metasploit crew.

Simple, enter the IP ranges like you would in NMap i.e. 192.168.1.-254 or 192.168.1.1/24 or whatever you want and it'll run against those hosts. Additionally you can place NMAP commands within the autopwn ip ranges bar, for example, if you want to scan even if a host "appears down" just do -PN 192.168.1.1-254 or whatever...you can use all NMap syntaxes in the Autopwn IP Ranges portion.

When it has completed exploiting simply type this:

```
sessions -l (lists the shells spawned)
sessions -i <id> (jumps you into the sessions)
```

```
Example 1: -PN 192.168.1.1
Example 2: 192.168.1.1-254
Example 3: -P0 -v -A 192.168.1.1
Example 4: 192.168.1.1/24
```

Enter the IP ranges to autopwn or (q)uit FastTrack: 163.117.132.169

Do you want to do a bind or reverse payload?

Bind = direct connection to the server  
Reverse = connection originates from server

1. Bind
2. Reverse

Enter number: 2

Launching MSFConsole and prepping autopwn...

```
db_driver postgresql
db_nmap 163.117.132.169
db_autopwn -p -t -e -r
sleep 5
jobs -K
```

```
sessions -l
echo "If it states No sessions, then you were unsuccessful. Simply type
sessions -i <id> to jump into a shell"
[*] The initial module cache will be built in the background, this can take 2-5
minutes...
```

```
IIIIII dTb.dTb      _.-.-._
II      4'  v  'B    .'"'.'/'|\'.'"'.
II      6.      .P   :  .' / | \ . :
II      'T; . ;P'   '.' / | \ . '
II      'T; ;P'    \. / | \ . '
IIIIII  'YvP'      \.-_|_|.-'
```

I love shells --egypt



```
= [ metasploit v4.5.0-dev [core:4.5 api:1.0]
+ -- --=[ 927 exploits - 499 auxiliary - 151 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops
```

```
msf > db_driver postgresql
[-] The db_driver command is DEPRECATED
```

Because Metasploit no longer supports databases other than the default PostgreSQL, there is no longer a need to set the driver. Thus db\_driver is not useful and its functionality has been removed. Usually Metasploit will already have connected to the database; check db\_status to see.

```
[*] postgresql connected to msf3dev
msf > db_nmap 163.117.132.169
[*] Nmap: Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-10-03 16:10 EDT
[*] Nmap: Nmap scan report for pc-132-169.uc3m.es (163.117.132.169)
[*] Nmap: Host is up (0.059s latency).
[*] Nmap: Not shown: 845 closed ports, 149 filtered ports
[*] Nmap: PORT      STATE SERVICE
[*] Nmap: 80/tcp    open  http
[*] Nmap: 443/tcp   open  https
[*] Nmap: 1042/tcp  open  afrog
[*] Nmap: 3306/tcp  open  mysql
[*] Nmap: 3389/tcp  open  ms-wbt-server
[*] Nmap: 10000/tcp open  snet-sensor-mgmt
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 12.55 seconds
msf > db_autopwn -p -t -e -r
[-] The db_autopwn command is DEPRECATED
[-] See http://r-7.co/xY65Zr instead
msf > sleep 5
msf > jobs -K
Stopping all jobs...
msf >
msf >
msf >
msf >
msf > sessions -l
```

```
Active sessions
=====
```

No active sessions.

```
msf > echo "If it states No sessions, then you were unsuccessful. Simply type
sessions -i <id> to jump into a shell"
```

```
[*] exec: echo "If it states No sessions, then you were unsuccessful. Simply
type sessions -i <id> to jump into a shell"
```

```
If it states No sessions, then you were unsuccessful. Simply type sessions -i
<id> to jump into a shell
msf >
```

Como podemos observar, Fast-Track no ha conseguido penetrar en el servidor.

## Capítulo 8

### Conclusiones

Partiendo de los objetivos planteados podemos concluir que el sistema desarrollado permite realizar predicciones de precios horarios, mensuales y pico de forma estable y pone los resultados a disposición de los usuarios con una interfaz sencilla e intuitiva.

Para el desarrollo de la misma se han utilizado tecnologías y lenguajes de programación que hoy en día representan, probablemente, la arquitectura de servicios más extendida dada su potencia, flexibilidad para adaptarse a cualquier entorno y reducido coste relativo a licencias.

Siguiendo como principio la máxima reducción en coste posible, ha sido necesario consultar las diferentes tecnologías que se están utilizando actualmente en el desarrollo de aplicaciones Web, distribuidas bajo una licencia pública, como PHP, Java, MongoDB, MySQL, etc.

He adquirido los conocimientos de programación en PHP y Java, haciendo especial énfasis en las capacidades que estas pueden tener, he utilizado JavaScript junto con HTML/XHTML y he aprendido a utilizar un gestor de bases de datos NoSQL como MongoDB.

Cabe destacar que he optado por el driver ODBC para MySQL y no por el software desarrollado por mí, ya que, aunque se obtengan menores velocidades de acceso y consulta, el driver ODBC es más seguro y es una vía menos susceptible de ser atacada. Dejo a disposición libre la pasarela MySQL-PHP desarrollada.

En cambio, para el servidor HTTP, he optado por usar Apache Web Server, ya que los resultados de stress muestran una mayor robustez. Ello se debe a que PriceForecast Web Server, a diferencia de Apache, requiere la ejecución de PHP en modo FastCGI y éste no está completamente desarrollado para Windows. En este entorno, PHP es incapaz de manejar un elevado número de peticiones simultáneas.



Esta la principal razón por la que PriceForecast y el resto de servidores web como Nginx y Lighttpd pueden ejecutarse en Windows pero su uso en entornos de producción está especialmente desaconsejado. Dejo a disposición libre el código de PriceForecast Web Server.

Uno de los objetivos iniciales del proyecto era el aseguramiento de la robustez y la seguridad del software desarrollado. Para ello, el sistema ha sido sometido a un plan de pruebas definido para este fin. Los resultados han sido satisfactorios y por ello se puede concluir que el software cumple con las funcionalidades para las que fue diseñado.

## Capítulo 9

### Bibliografía

- [1] *Booch, G.* UML: el lenguaje unificado de modelado: guía del usuario. Editorial: Addison-Wesley. 2ª Edición. ISBN: 8478290761.
- [2] *Russ M.* Learning UML 2.0. Editorial: O'Reilly Media. 1ª Edición. ISBN: 0596009828.
- [3] *Fawcett J.* Beginning XML. Editorial: Wrox. 5ª Edición. ISBN: 1118162137.
- [4] *Ray, E.* Learning XML. Editorial: O'Reilly Media. 2ª Edición. ISBN: 05960004206.
- [5] *Duckett J.* Beginning HTML, XHTML, CSS and JavaScript. Editorial: Wrox. 1ª Edición. ISBN: 0470540702.
- [6] *Felke-Morris T.* Web Development and Design Foundations with XHTML. Editorial: Addison-Wesley. 5ª Edición. ISBN: 0132122707.
- [7] *Attaway, S.* A Practical Introduction to Programming and Problem Solving. Editorial: Butterworth-Heinemann. 2ª Edición. 0123850819.
- [8] *Hahn, B.* Essential Matlab for Engineers and Scientists. Editorial: Academic Press. 4ª Edición. ISBN: 0123748836.
- [9] *Altman, Y.* Undocumented Secrets of Matlab – Java Programming. Editorial: Chapman and Hall. 1ª Edición. ISBN: 1439869030.
- [10] *Davis, T.* MATLAB Primer. Editorial: CRC Press. 8ª Edición. ISBN: 1439828628.
- [11] *Bloch, J.* Effective Java. Editorial: Addison-Wesley. 2ª Edición. ISBN: 0321356683.
- [12] *Sarang, P.* Java Programming (Oracle Press). Editorial: McGraw-Hill Osborne Media. 1ª Edición. ISBN: 007163360X.
- [13] *Harold, E.* Java Network Programming. Editorial: O'Reilly Media. 3ª Edición. ISBN: 0596007213.
- [14] *Welling, L.* PHP and MySQL Web Development. Editorial: Addison-Wesley. 4ª Edición. ISBN: 0672329166.
- [15] *Beaulieu, A.* Learning SQL. Editorial: O'Reilly Media. 2ª Edición. ISBN: 0596520832.

- [16] *Morimoto, R.* Microsoft Windows Server 2003 Unleashed (R2 Edition). Editorial: Sams Publishing. ISBN: 0672328984.
- [17] W3C. <http://www.w3c.org/protocols>. Abril 2012.
- [18] OMI, Polo Español, S.A. (OMIE) <http://www.omie.es>. Septiembre 2011.
- [19] Red Eléctrica de España, S.A. <http://www.ree.es>. Noviembre 2011.
- [20] Sistema de Información ESIOS. <http://www.esios.ree.es>. Noviembre 2011.
- [21] Energía y Sociedad. Campus Iberdrola. <http://www.energiaysociedad.es>. Agosto 2012.
- [22] TechCenter de seguridad. <http://technet.microsoft.com>. Septiembre 2012.
- [23] Nmap – Free Security Scanner for Network Exploration & Security Audits. <http://www.nmap.org>. Septiembre 2012.
- [24] Nessus Scanner. <http://www.tenable.com>. Septiembre 2012.
- [25] Apache. <http://httpd.apache.org>. Septiembre 2012.
- [26] MySQL. <http://www.mysql.org>. Noviembre 2011.
- [27] MySQL Dev. <http://dev.mysql.org>. Agosto 2012.
- [28] BackTrack Linux – Penetration Testing Distribution. <http://www.backtrack-linux.org>. Agosto 2012.



## Anexo I

### Planificación

En este anexo se expondrá un diagrama de GANTT con la planificación del proyecto. El diagrama de GANTT es una herramienta que permite al usuario modelar la planificación de las tareas necesarias para la realización de un proyecto. La lectura de los diagramas de GANTT es relativamente sencilla y constituye una de las herramientas básicas en cualquier rama de la ingeniería.



ID	Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin
1		Inicio del proyecto	0 días	mié 03/11/10	mié 03/11/10
2		Introducción al proyecto	3 días	mié 03/11/10	vie 05/11/10
3		<b>Estado del arte</b>	<b>25 días</b>	<b>lun 08/11/10</b>	<b>vie 10/12/10</b>
4		Documentación del mercado eléctrico español	2 sem.	lun 08/11/10	vie 19/11/10
5		Documentación información pública	1 sem	lun 22/11/10	vie 26/11/10
6		Análisis de la herramienta Matlab de previsión de precios	2 sem.	lun 29/11/10	vie 10/12/10
7		<b>Tecnologías aplicadas y herramientas de trabajo</b>	<b>210 días</b>	<b>lun 13/12/10</b>	<b>vie 30/09/11</b>
8		Sistema operativo	1 sem	lun 13/12/10	vie 17/12/10
9		Servidores web	3 sem.	lun 20/12/10	vie 07/01/11
10		Sistemas de gestión de bases de datos	4 sem.	lun 10/01/11	vie 04/02/11
11		Matlab	3 sem.	lun 07/02/11	vie 25/02/11
12		XHTML & XML	2 sem.	lun 28/02/11	vie 11/03/11
13		<b>Lenguajes de programación</b>	<b>120 días</b>	<b>lun 14/03/11</b>	<b>vie 26/08/11</b>
14		JavaScript	1 ms	lun 14/03/11	vie 08/04/11
15		PHP	2 mss	lun 11/04/11	vie 03/06/11
16		Java	3 mss	lun 06/06/11	vie 26/08/11
17		Documentación Web Server Stress Test	5 días	lun 29/08/11	vie 02/09/11
18		Documentación Web-Based Exploitation	1 sem	lun 05/09/11	vie 09/09/11
19		Documentación Port & Vulnerability Scanning	1 sem	lun 12/09/11	vie 16/09/11
20		Documentación Server Exploitation	2 sem.	lun 19/09/11	vie 30/09/11
21		<b>Backend</b>	<b>105 días</b>	<b>lun 03/10/11</b>	<b>vie 24/02/12</b>
22		Análisis	1 ms	lun 03/10/11	vie 28/10/11
23		Diseño	5 sem.	lun 31/10/11	vie 02/12/11
24		Implementación	3 mss	lun 05/12/11	vie 24/02/12
25		<b>Frontend</b>	<b>80 días</b>	<b>lun 27/02/12</b>	<b>vie 15/06/12</b>
26		Análisis	3 sem.	lun 27/02/12	vie 16/03/12
27		Diseño	5 sem.	lun 19/03/12	vie 20/04/12
28		Implementación	2 mss	lun 23/04/12	vie 15/06/12
29		<b>API</b>	<b>30 días</b>	<b>lun 18/06/12</b>	<b>vie 27/07/12</b>
30		Diseño	1 sem	lun 18/06/12	vie 22/06/12
31		Implementación	3 sem.	lun 25/06/12	vie 13/07/12
32		Aplicación cliente	10 días	lun 16/07/12	vie 27/07/12
33		<b>Evaluación</b>	<b>6 días</b>	<b>lun 30/07/12</b>	<b>lun 06/08/12</b>
34		Web Server Stress Tool	1 día	lun 30/07/12	lun 30/07/12
35		Web-based Exploitation	2 días	mar 31/07/12	mié 01/08/12
36		Port & Vulnerability Scanning	2 días	jue 02/08/12	vie 03/08/12
37		Server Exploitation	1 día	lun 06/08/12	lun 06/08/12
38		Documentación	9 sem.	mar 07/08/12	lun 08/10/12

The Gantt chart visualizes the project schedule. Tasks are represented by horizontal bars indicating their duration. Dependencies are shown as arrows connecting tasks. Key milestones include the start of the project (Nov 3, 2010), completion of the State of the Art (Dec 10, 2010), completion of Backend development (Feb 24, 2012), and completion of Frontend development (Jun 15, 2012). The final evaluation phase concludes on August 6, 2012.

Proyecto: Proyecto.mpp  
Fecha: lun 08/10/12

<div><div></div> Tarea</div> <div><div></div> Resumen</div> <div><div></div> Resumen del proyecto</div> <div><div></div> Tareas externas</div>	<div><div></div> Hito externo</div> <div><div></div> Tarea inactiva</div> <div><div></div> Hito inactivo</div>	<div><div></div> Resumen inactivo</div> <div><div></div> Tarea manual</div> <div><div></div> Sólo duración</div>	<div><div></div> Informe de resumen manual</div> <div><div></div> Resumen manual</div> <div><div></div> Sólo el comienzo</div>	<div><div></div> Sólo fin</div> <div><div></div> Fecha límite</div> <div><div></div> Progreso</div>
--	--	--	--	---

Página 1

## Anexo II

### Presupuesto

En este anexo se analizan los costes que ha generado la elaboración del proyecto, teniendo en cuenta el coste de personal, así como las licencias de programas informáticos, equipos y material.

#### 11.1 Coste de personal

Uno de los costes más relevantes para el presupuesto del proyecto es imputable al personal. Debemos calcular el coste total de personal en función de las horas que se han dedicado al proyecto. Para la elaboración del proyecto se han requerido un total de 504 días. Asignando cuatro horas de trabajo diario, el número total de horas trabajadas son 2016.

Para el cálculo del coste de personal, tenemos que partir de la titulación del trabajador. En este caso, el personal encargado del proyecto es un Ingeniero Industrial de categoría Junior con un coste de 1900€ al mes. El cálculo final se muestra en la siguiente tabla.

Cualificación Personal	Categoría	Dedicación [horas]	Coste [€/h]	Coste [€]
Ingeniero Industrial	Ingeniero Junior	2016	15,31	30.864,96
			<b>TOTAL</b>	30.864,96

Tabla 41. Coste de personal

## 11.2 Hardware

La siguiente tabla muestra el coste de los equipos informáticos adquiridos durante el periodo de duración del proyecto.

Descripción	Unidades	Coste unitario [€]	Coste Total [€]
Servidor Intel Core 2 Quad CPU	1	1,795.99	1,795.99
		<b>TOTAL</b>	1,795.99

Tabla 42. Detalle de hardware

## 11.3 Software y Licencias

La siguiente tabla muestra el coste de los equipos informáticos adquiridos durante el periodo de duración del proyecto.

Descripción	Unidades	Coste unitario [€]	Coste Total [€]
Windows Server 2003	1	349.95	349.95
IDE Eclipse	1	0	0
IDE Aptana	1	0	0
Apache Server	1	0	0
Servidor MySQL	1	0	0
Servidor MongoDB	1	0	0
Librerías externas	8	0	0
Herramientas de análisis	5	0	0
		<b>TOTAL</b>	349.95

Tabla 43. Coste software y licencias

## 11.4 Resumen

Sumando los costes asociados a personal, equipos y licencias, obtenemos un coste total de 33.010,90 €.

Concepto	Coste [€]
Personal	30.864,96
Hardware	1,795.99
Software y licencias	349.95
<b>TOTAL</b>	<b>33,010.90</b>

Tabla 44. Resumen costes

